DOCUMENT RESUME

ED 244 636 IR 050 735

TITLE	(Networking + Integrating) * (Systems + Society). Proceedings of the Annual Canadian Conference of Information Science (12th, Toronto, Ontario, May 14-16, 1984) = (Reseaux + Integration) * (Systemes + Societe). Comptes rendus de la conference annuelle Canadienne des sciences de l'information (12th, Toronto, Ontario, 14-16 mai, 1984).
INSTITUTION	Canadian Association for Information Science, Ottawa (Ontario).
PUB DATE	May 84
NOTE	205p.
AVAILABLE FROM	CAIS Secretariat, Sonoptic Communications, Suite 100, 44 Bayswater Avenue, Ottawa, Ontario, Canada K1Y 4K3 (Members: $25.00; non-members: $35.00).
PUB TYPE	Collected Works - Conference Proceedings (021) -- Viewpoints (120) -- Reports - General (140)
LANGUAGE	English; French
EDRS PRICE	MF01/PC09 Plus Postage.
DESCRIPTORS	Cataloging; Computer Software; Databases; Futures (of Society); *Information Dissemination; *Information Networks; *Information Retrieval; Information Science; Information Storage; Library Services; Microcomputers; *Online Systems; Professional Associations; *Technological Advancement; User Satisfaction (Information)
IDENTIFIERS	Universal Relation Theory

ABSTRACT

Seventeen papers from the 1984 annual conference of the Canadian Association for Information Science (CAIS) are presented in four broad topic areas. The first group, which focuses on changing roles in information access, includes the keynote address by Charles Meadow, "Integrating Access to Information Utilities: Promises, Problems, and Profiles of the Future." Other papers in this area are concerned with office automation, the role of the corporate database administrator, and database generation by end-users. The two papers in the second group provide a discussion of the potential impact of fifth-generation computers on library and information services, and a description of the proposed computerization of document management in a specialized center in Quebec. Focusing on systems and software in networking, papers in the third section address the application of Universal Relation Theory to bibliographic design, documents as arrays, costs of query processing in a distributed database, advantages and disadvantages of microcomputer systems for acquisitions, one-time use programs for special database applications, the CAIS Information Technology Special Interest Group (ITSIG), and the integration of Telidon with BASIS. Papers in the fourth section focus on the impact of information technology on society. Topics include informatics (the science of information technologies) and information dependency, communications systems for physicians, a network for occupational health and safety in Quebec, and individual privacy versus corporate and governmental information systems. The two papers describing programs in Quebec are in French, the remainder in English. Abstracts in both English and French are included for each paper, and an author index is provided. (DMC)

ED244636

U.S. DEPARTMENT OF EDUCATION
NATIONAL INSTITUTE OF EDUCATION
EDUCATIONAL RESOURCES INFORMATION
CENTER (ERIC)

- This document has been reproduced as received from the person or organization originating it.
- Minor changes have been made to improve reproduction quality.
- Points of view or opinions stated in this document do not necessarily represent official NIE position or policy.

PROCEEDINGS
OF THE
TWELFTH ANNUAL
CANADIAN CONFERENCE
ON INFORMATION SCIENCE

COMPTES RENDUS
DE LA
DOUZIÈME CONFÉRENCE
ANNUELLE CANADIENNE DES
SCIENCES DE L'INFORMATION

(NETWORKING
+
INTEGRATING)
*
(SYSTEMS
+
SOCIETY)

(RÉSEAUX
+
INTÉGRATION)
*
(SYSTÈMES
+
SOCIÉTÉ)

TRIUMPH SHERATON HOTEL
TORONTO, ONTARIO
MAY 14 - 16 MAI, 1984

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY
Ethel Auster
TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

CANADIAN ASSOCIATION
FOR INFORMATION SCIENCE

ASSOCIATION CANADIENNE
DES SCIENCES DE
L'INFORMATION

IR050735

Copyright © 1984 by
Canadian Association
for Information Science

ALL RIGHTS RESERVED

Can be purchased from
Canadian Association for
Information Science:
CAIS Secretariat
Sonoptic Communications
Suite 100
44 Bayswater Avenue
Ottawa, Ontario
K1Y 4K3

Price:	Members:	$25.00
	Non-members:	$35.00

Legal deposit:
National Library of Canada

Copyright © 1984 par
l'Association canadienne
des sciences de l'information

TOUS DROITS RESERVÉS

En vente à
l'Association canadienne
des sciences de l'information:
Secrétariat de l'ACSI
Sonoptic Communications
Suite 100
44 Bayswater Avenue
Ottawa, Ontario
K1Y 4K3

Prix:	Membres:	$25.00
	Non-membres:	$35.00

Dépôt légal:
Bibliothèque nationale
du Canada

3

CAIS CONFERENCE 1984/CONFÉRENCE L'ACSI 1984

CONFERENCE ORGANIZATION COMMITTEE/COMITÉ ORGANISATEUR DE LA CONFÉRENCE

CHAIRPERSON/PRÉSIDENTE:	Ethel Auster
EXHIBITS/EXPOSITION:	Rick Noble
FINANCE:	Ann Schabas
PROGRAMME:	Gordon Bean Sally Grande
PUBLICATIONS:	Paul Gallina Susan Cozzi
PUBLICITY/PUBLICITÉ:	Ruth von Fuchs Ann Poole
REGISTRATION/INSCRIPTION:	Barbara Fingerote Felicity Pickup Jim Brownridge
SECRETARY/SECRÉTAIRE:	Mary Campbell
SPONSORSHIP/COMMANDITAIRES:	John Campbell
TRANSLATOR/TRADUCTEUR:	Gilles Deschatelets

TABLE OF CONTENTS/TABLE DES MATIÈRES

CONFERENCE ORGANIZATION COMMITTEE/COMITÉ ORGANISATEUR DE LA CONFÉRENCE iii

CHANGING ROLES IN INFORMATION ACCESS/TRANSFORMATION DES RÔLES DANS L'ACCESSIBILITÉ À L'INFORMATION

KEYNOTE ADDRESS/CONFÉRENCE D'OUVERTURE

INTEGRATED ACCESS TO INFORMATION UTILITIES: PROMISES, PROBLEMS AND PROFILES OF THE FUTURE/L'ACCÈS INTÉGRÉ AUX SOURCES D'INFORMATION: PROMESSES, PROBLÈMES ET PROFIL DE L'AVENIR 1
Charles T. Meadow

PILOTING OFFICE SYSTEMS-A USER'S PERSPECTIVE/LE PILOTAGE DES SYSTÈMES DE BUREAU-POINT DE VUE DE L'USAGER 9
T. Grusec, N. Park

THE EMERGING CORPORATE DATABASE ADMINISTRATOR/UNE NOUVELLE FONCTION: LA GESTION CORPORATIVE DES BASES DE DONNÉES 23
A.W. Johnston, J.L. Barker

DATABASE GENERATION BY END-USERS/CRÉATION DYNAMIQUE DE BASES DE DONNÉES PAR LES UTILISATEURS 36
Raymond G. Blackie

NIGHT FLIGHTS/VOLS DE NUITS

THE IMPACT OF FIFTH-GENERATION COMPUTERS ON LIBRARY AND INFORMATION SERVICES/L'IMPACT DES ORDINATEURS DE CINQUIÈME GÉNÉRATION SUR LES BIBLIOTHÈQUES ET LES CENTRES DE DOCUMENTATION 45
Michael A. Shepherd

"GESTDOC", OU L'INFORMATISATION DE LA GESTION DE DOCUMENTS À L'INTÉRIEUR D'UN CENTRE DE DOCUMENTATION SPECIALISÉ/"GESTDOC", OR THE COMPUTERIZATION OF DOCUMENT MANAGEMENT WITHIN A SPECIALIZED DOCUMENTATION CENTRE 56
J.L. Benedetti, L. Lefebvre, G. Sanfaçon, J.P. Weber, S. Gallichand

INTEGRATING AND NETWORKING: SYSTEMS AND SOFTWARE/
RÉSEAUX ET INTÉGRATION: SYSTÈMES ET LOGICIELS

APPLYING UNIVERSAL RELATION THEORY TO BIBLIOGRAPHIC DESIGN/ THÉORIE DE LA RELATION UNIVERSELLE APPLIQUÉE À LA CONCEPTION BIBLIOGRAPHIQUE 70
R.G. Crawford

DOCUMENTS AS ARRAYS/REPRÉSENTATION MATRICIELLE DES DOCUMENTS DANS LES BASES DE DONNÉES RELATIONELLES 77
Ian MacLeod

COST FUNCTIONS FOR QUERY PROCESSING IN A DISTRIBUTED DATABASE/ LES COMPOSANTES DU COÛT DES RECHERCHES SUR DES BASES DE DONNÉES RÉPARTIES 89
P. Bodorik

REWARDS AND PITFALLS OF TURNING TOWARD A MICRO (AND AWAY FROM INTEGRATED SYSTEMS) FOR ACQUISITIONS/AVANTAGES ET INCONVÉNIENTS DU MICROODINATEUR (EN DEHORS D'UN SYSTÈME INTÉGRÉ) POUR LES ACQUISITIONS 114
Mary E. Campbell, L.F. "Buddy" Doll

JUSTIFYING ONE-TIME USE PROGRAMS FOR SPECIAL DATABASE APPLICATIONS/L'UTILISATION DE LOGICIELS JETABLES POUR CERTAINES BASES DE DONNÉES PARTICULIÈRES 133
John C. Nash, Mary M. Nash

CAIS INFORMATION TECHNOLOGY SPECIAL INTEREST GROUP (ITSIG)/ LE GROUPE D'INTÉRÊT SPÉCIAL SUR LA TECHNOLOGIE DE L'INFORMATION DE L'ACSI (ITSIG) 142
Roddy Duchesne

INTEGRATION OF TELIDON WITH BASIS - A QUERY-DRIVEN BIBLIOGRAPHIC UTILITY/L'INTEGRATION DE TÉLIDON ET DE BASIS- UN SYSTÈME BIBLIOGRAPHIQUE DE TYPE "QUESTION-RÉPONSE" 150
Charles P. Cohen

INTEGRATING AND NETWORKING: TECHNOLOGY AND SOCIETY/
RÉSEAUX ET INTEGRATION: TECHNOLOGIE ET SOCIÉTÉ

INFORMATICS AND INFORMATION DEPENDENCY/L'INFORMATIQUE ET LA DÉPENDANCE DE L'INFORMATION 156
Victor M. Owen

PHYSICIANS' COMMUNICATIONS SYSTEMS: COPING WITH INFORMATION OVERLOAD/DES SYSTÈMES DE COMMUNICATION POUR MÉDICINS: VENIR À BOUT DES SURCHARGES D'INFORMATION 163
Joanne L. Warren

LE RÉSEAU ISST: SANTÉ ET SÉCURITÉ DU TRAVAIL AU QUÉBEC/THE ISST NETWORK: OCCUPATIONAL HEALTH AND SAFETY IN QUEBEC 178
Suzanne L'Heureux, Jean Pierre Arsenault

MIPS TO THE PEOPLE/MIPS AUX GENS 190
Richard Larratt

AUTHOR INDEX/INDEX DES AUTEURS 197

INTEGRATED ACCESS TO INFORMATION UTILITIES: PROMISES, PROBLEMS AND PROFILES OF THE FUTURE

Charles T. Meadow
DIALOG Information Services, Inc.
Palo Alto, California 94304, U.S.A.

ABSTRACT

This paper reviews, in a historical context, how some of the access techniques in current usage came about, what is the status of access techniques today, and what seem to be the immediate needs of users. The growing use of online information systems, and of user dependence on these systems accelerates the tendency to bring more information online and to improve access techniques.

8

L'accès intégré aux sources d'information: promesses, problèmes et profil de l'avenir.

Cette communication fait la révision, selon un contexte historique, de comment les techniques d'accès pour l'usage courant se font, quel est l'état des techniques d'accès à l'heure actuelle et que semblent être les besoins immédiats des utilisateurs. L'utilisation croissante de systèmes d'information «online» et de la dépendance des utilisateurs à ces systèmes accélère la tendance à produire plus d'information «online» et à améliorer les techniques d'accès.

INTEGRATED ACCESS TO INFORMATION UTILITIES: PROMISES, PROBLEMS AND PROFILES OF THE FUTURE

INTRODUCTION

I am going to review some of the history of access to online information services. Access is a process very much affected by available technology and the behavior of its users. I am not going to review content, a wholly separate issue, even though changes in access technology affect the amount, quality and type of information that providers are motivated to put into online systems. Also, of course, the amount, quality and type of information available affects the behavior of users (how much they are willing to pay, for example) and the amount, quality and type of access systems entrepreneurs will make available to users. This is not news, but to explore both sides of the issue would take the entire week.

1. SOME HISTORY OF ACCESS TO INFORMATION RETRIEVAL SYSTEMS

As a practical matter, the kinds of data access systems we talk of and use today had their genesis in the 1950's when they were not interactive. Users had to work out a 'query,' and have it conveyed to a computer in a batch processing mode. Turn around time was on the order of a day or more.

The very earliest languages were extremely complex, as designers strove to offer users the opportunity as well as the obligation of being quite precise, since they could not offer quick resolution of errors, or change of direction. It was common to use role indicators to modify terms (such designations as product, process, material, ...) and various kinds of linkage codes for connecting terms (A located in B, A acting upon B, ...). Basically, these were attempts to reproduce much of the syntax of natural language without its redundancy and context dependency. But, redundancy and context dependency, while servng to confuse the reader also enable the writer to be creative in expression. In poetry, we use old words, in new way, to create new images. Instead of teaching the computer how to understand natural language, we tried in those days to teach the human users how to deal with a new, artificial, and cumbersome language. The computer cannot say "If I have to learn an uncomfortable new language to use your system, I simply will not use it." But, humans can say that, and do. And so, by and large, cumbersome languages, however logically powerful, do not sell well with the general user population.

Users in those days were few in number. Even fewer were users who were dependent on the systems. This is important: there was no habit of the use of information retrieval systems. Users either accepted the limitations imposed, or stayed away -- mostly the latter.

In the late 1960's to early 1970's, time-sharing computers and relatively inexpensive data communications networks were developed. They, together with some good software design, made online information retrieval a practical reality.

The speed of response improved immensely: a single command might be responded to in seconds. A complete query might be sent to the computer, tested, revised and run to completion in 15 minutes.

The languages of use simplified. Basically, as quick reaction became the order of the day, it was better to abandon the highly complex languages of the batch systems in favor of simple languages, whose statements could be easily and quickly tested. The idea of 'conversing' with a computer was born, not only in the information retrieval context, but also in computer programming and computer graphics. Basically, conversation means one can say something and see an almost immediate reply -- the effect of one's utterance. This is what we mean by human conversation and this is the process from which my company took its name -- immediate interaction compared with, say, making a speech or lecture in which the effect of one's utterances may not be felt for some time. With a truly interactive system, one can program a query as a series of probing steps, with confirmation at each step that the entire process seems to be going right.

Up to about about a year ago, our situation was that speed had improved and communication was cheaper and more reliable. Languages were growing somewhat more complex, as we added new functions and new variations on old ones. Users were becoming sophisticated. They were beginning to recognize that there were some capabilities they still did not have.

The typical user was still an information professional. It still took a day or two of instruction for a person who understood the information and its structure to learn how to search the huge files that were appearing and growing.

2. THE SITUATION TODAY

Today, the effect of microcomputers is strikingly apparent. The proportion of new users of systems like DIALOG are about half individuals, rather than corporations or libraries. Communication speed continues to increase: 1200 baud is common, 2400 is now commercially available.

Systems are available world-wide. Users with reasonable financial support are nearly independent of the country of origin of their target systems.

Attention is being paid to the inexperienced user, especially to the user who is a busy professional in a field other than information science and who will simply not take the traditional path to learning how to use complex programs. To support such users, we see systems like Scimate, Search Helper, and INSEARCH which, each in its own way, help the user to perform a search, eliminating much of the burden for him to learn the intricate details of search language and syntax and of database structure.

The 'window' is upon us. Basically, this is a technique for splitting the screen of a CRT, allowing the user to see several displays at once, like overlayed pieces of paper on his desk. The advantage is that complex problems often require simultaneous attention to several sets of data: a sample of retrieved output, a search history, the most recent error message, the list of choices for next action, ... This technique is made possible by high resolution CRT screens and some clever programming.

sets of data: a sample of retrieved output, a search history, the most recent error message, the list of choices using the output of one retrieval operation directly in the next, as 'Find the CAS registry numbers of substances with these trade names: <...list...> and then retrieve citations concerning them,' or 'Find articles about <...list of terms...> and order the resulting set by frequency of occurrence of the terms to provide an approximation to relevance ordering.' For example, one might use word or descriptor frequencies to find the records of a set richest in the most interesting terms, thereby to drop off those that are less interesting. Another possibility would be to order a set of patent records, in a given technology, by assignee, thereby producing a list of the most active companies in the selected field.

To summarize our present situation, we have:

1. High speed communication, so that the time we have to wait for some action to be completed is becoming less and less a factor in searching time or economics.

2. Languages are becoming richer, enabling the knowledgable user to be more precise in selection criteria or to get more work done per command.

3. Alternatively, for the user who is not skilled and does not want to invest the time to become so, we are seeing a new level of simple languages evolve. These may be simple command languages, menu systems, or other techniques for accomplishing work with little prior learning of search mechanics.

4. Greater resolution in display screens permits use of windows which provide more then one visual field in the display, such as a word list, a few retrieved records, a search history and a list of possible next commands. These might benefit users at all skill levels.

5. We have also achieved a very significant step in user behavior. The typical user is now dependent on the online information system. This has implications to cost and reliability. Costs in this industry have generally not risen with inflation. Technologically enabled productivity increases have aided the consumer whose relative costs are holding steady or even decreasing. Reliability has also increased. Neither the computers not the communication systems are 'down' as often as they once were. At DIALOG, ninety eight percent is actually our desirable target. Again, there is circularity here: higher demand calls for higher reliability and higher reliability fosters user dependence, hence demand.

3. WHAT IS STILL MISSING?

It is no surprise that the combined computer-communications industry, ever producing new capabilities and raising consumer expectations for still more, leaves us today with some unfulfilled desires. We can see capabilities in related fields and want to have them in our field as well.

Some of these are:

1. True multi-media output. Audio, in effect the 'voice over' of television, can be used to provide control and error information to the user, reserving the visual field for substantive information. I believe it would be both less distractive and more pleasing to the user to have more than one sense involved in receiving computer output. Audio could also be applied to aid the visually handicapped receive any kind of computer output. Video can be used in training, to increase the attentiveness of the user. It can also be used to transmit the pictorial part of a stored record. High resolution graphics would be an invaluable adjunct to the text of online scientific papers, patent documents or trademark records.

2. More active assistance to the users. We want not just instruction, or a passive reduction in complexity of language, but an active capability on the part of the search system to provide some comprehension of what the user is trying to do, or has done wrong, and to offer active help in proceeding.

3. Linkage of retrieval systems with processing systems, such as word processors or statistical, graphical or spread sheet systems. Retrieval, per se, is rarely the user's objective. The information to be retrieved is wanted as part of some next step. Why not provide the linkage directly? We are just beginning to see some of these come about, although there does not seem to be high use yet.

4. Direct problem-solving software. If a user wants a profile of a company or of a country's economy, or information about the side effects of a drug, why not ask the system essentially that question? I do not suggest that we need to go to the extent of full computer comprehension of natural language, certainly not to the extent of the talking, wise-cracking retrieval systems of the science fiction movies, but simply that we provide whole libraries of problem-oriented information retrieval software. These programs would be analogous to the specialized software we now see for budgeting, inventory control, bank account balancing, and the like, which in a sense replace general database management systems for the problem oriented user. The problem oriented user is one who is mainly concerned with the content of the results, as contrasted with a process oriented user whose main interest is the manner in which a result is obtained.

4. PROSPECTS

Naturally, I feel that we shall soon see all the items I listed as lacking. As has always been true in computing, we shall also have developments which we do not foresee, and they may come to dominate the ways in which we use information systems. For example, few people were predicting, say ten years ago, the extent of the development and use of personal computers, which today seem to be revolutionizing the information seeking world.

I truly believe that information retrieval, like mathematics or writing, is a part of almost all other human endeavors, certainly those associated with management, research or technology. Everyone will use retrieval systems and everyone will have to know how. Systems will have to be effective, for many types of users. That is partly a challenge to education and partly to software designers and information providers.

PILOTING OFFICE SYSTEMS - A USER'S PERSPECTIVE

T. Grusec, Ph.D., Associate Director,
OCS Program, Dept. of Communications,
Government of Canada, Ottawa, K1A 0C8

and

N. Park, Ph.D., Senior Evaluator,
OCS Program, Dept. of Communications,
Government of Canada, Ottawa, K1A 0C8

ABSTRACT

An overview of the Office Communications Systems (OCS) Program within the Department of Communications of the Federal Government is presented. The three aims of the program-industry stimulus, government experience and impact assessment - are discussed along with their embodiment in four field trials of integrated office technologies addressed at multi-function knowledge workers in four different government departments. The trials are a learning experience for all concerned. The aims of impact assessment are outlined. Although results are only now beginning to come in, the experience to date with impact assessment has contributed to a clear framework from users' points of view of the critical steps which should be followed for successful implementation of these office technologies. Five steps are presented with stress given to needs analysis from users' points of view as the initial and most important step from which the other ones follow.

Le pilotage des systèmes de bureau - Point de vue de l'usager

Un examen du Programme du système de communications de bureau du ministère fédéral des Communications est présenté. Les trois buts du programme - stimuler l'industrie, donner l'expérience aux services du gouvernement et évaluer l'impact du système - sont présentés, ainsi que leur incorporation dans quatre domaines d'essai de technologies de bureau intégrées s'adressant au personnel des connaissances aux fonction multiples dans quatre services gouvernementaux différents. Les essais sont une bonne expérience pour ceux qui sont concernés. Les buts de l'évaluation de l'impact sont étudiés. Bien que les résultats commencent seulement à revenir, on peut déjà voir comment cette évaluation a contribué à une meilleure perception, de la part de l'usager, des opérations importantes à suivre pour introduire avec succès ces technologies de bureau. Cinq étapes sont proposées, en mettant l'accent, du point de vue de l'usager, sur l'analyse des besoins qui doit être la première étape la plus importante et après laquelle suivent les autres étapes.

PILOTING OFFICE SYSTEMS

INTRODUCTION

The Office Communications Systems (OCS) Program is a joint industry/ Canadian Federal Government initiative administered by the Department of Communications. The program has three aims. Its basic objective is to promote the development of a Canadian industrial capacity for supplying integrated office systems to domestic and world markets. By providing selected federal government departments as test beds for technology, a second aim, government experience with such systems, is achieved. Thirdly, the mandate of the program calls for research within the test departments on the psychological, social and organizational impacts of the technologies.

INDUSTRY STIMULUS

The program's basic aim derives from present and forecast Canadian trade deficits in office technologies. A study by Arthur D. Little, Inc., showed that the world market for office products in 1983 was about $100 billion, and that for Canada was about $5 billion. These figures were projected to double by 1990. With Canada's share of the world market at $1 billion and of the domestic market at $0.7 billion in 1983, a net trade deficit of $3 billion resulted in that year. This deficit could more than double by 1990 if present trends continue and this represents a great deal of money by Canadian standards. The OCS Program was set up to aid in turning this deficit situation around by the stimulation of Canadian industrial effort.

GOVERNMENT EXPERIENCE

The kind of technologies being addressed by the OCS Program, integrated office systems, are primarily aimed at multi-function knowledge workers (managers and professionals). This segment of the workforce is quite costly in terms of salaries. The second goal of the Program, namely experience with integrated systems in government departments, is hoped to pave the way for increases in the efficiency and/or effectiveness of this expensive group of workers. Until now, computer technologies in the office have been primarily directed at highly structured support functions. Many plausible reasons can be put forth for the low penetration of the new technologies into knowledge work until now. These include, the lack of development of suitable technology configurations (except in very special instances), rejection by executives of keyboard operation, the absence of productivity measures to justify equipping this workforce segment. However, integrated systems are now believed to be a real growth area for automation. This is still a further reason for industrial stimulation of these particular technologies, as well as for obtaining experience in their use within government departments.

IMPACTS

Various scenarios of expected impacts of office technologies have been presented and widely publicized in the media. These range from doom and gloom to utopia. Beyond plausible speculation, however, little hard data exist to evaluate, and thence to anticipate and influence, the personal, social, organizational and societal effects of these technologies. Thus, the third aim of the Program is to gather sound evidence for appropriate evaluation of these scenarios by a thorough examination of events in the trials.

THE TRIALS

With funding of $13 million, these three Program objectives are embodied in four field trials, each one in a different government department with a different systems vendor. During the planning phase of the OCS Program, various committees and sub-committees were formed, and a number of systems vendors as well as a number of potential site departments came forth with strong interests in participation. It was felt that the best results could be obtained from multiple sites and vendors, rather than from one large trial, but not with so many sites as to dilute the scope of effort within any one trial. The four trials resulted after a careful matching of vendors' and site departments' interests and outlooks to their mutual satisfaction.

DIFFERENCES AMONG THE TRIALS

While all the trials are focused on multi-function knowledge work, they differ in interesting ways. These differences support the wisdom of spreading the OCS Program's dollars among several vendors and site departments since it cannot be foreseen which system approaches and which site department's applications will become winners in the Canadian and world marketplace.

On the technology side, the differences include interconnection via co-axial cable versus a fibre optic local area network versus a telephony-based twisted pair approach. Also, three of the trials are attempts at fully integrated systems. Such systems might include electronic mail, text editing, storage and retrieval, decision support, voice messaging, teleconferencing, calendaring, reminder services, and so on, all accessible from a single workstation.

The fourth trial is a more limited one with the software customized to entry, storage, update and retrieval of the policies and procedures of a government department. This smaller trial represents a different strategy from the other three. This is a gradual, stepwise strategy where a clearly delimited problem is solved in modular fashion, with the way open to adding modules aimed at other functions later on. But even in the three larger trials which, on the surface, appear similar, the details of how the software is worked out, how well it addresses the work functions at the site department, and how "user-friendly" it is, may produce quite different patterns and degrees of usefulness and acceptance. This in turn will help to determine the marketplace fate of each of these integrated systems.

On the applications side, we have already mentioned the work content of the smaller trial. But the three larger trials differ among each other as to which of the integrated functions are expected to provide major work effectiveness payoffs in their respective departments. At one site, emphasis is on data base access and report preparation, filing and retrieval; in another trial, decision support via spreadsheet capabilities, and financial management functions loom larger. Also, some of the trials are at a horizontal slice of a department, where similar knowledge work is done by a more-or-less homogeneous group, while another site covers a vertical spectrum from very senior management to technically oriented professionals.

All in all, these differences in technologies, approaches and applications among the trials, make for an interesting mix that promises to serve all three Program objectives much more effectively than would have been the case in a single large trial. Besides spreading the eventual market place risk among four systems vendors, the government experience is rendered considerably broader in the kinds of multi-function work addressed, and, consequently, a wider possible range of impacts is available for examination.

TIME SCALE

The first year of the Program was a planning phase at the end of which the vendors and site departments had been selected and matched. A needs analysis phase followed, taking up the next six months to one year at the various sites. Systems specification and development was next, also taking six months to one year. User training and initiation of the pilot phase with hands on testing is currently underway. The four trials are at varying points in this final pilot phase, with one almost completed, others barely begun. The OCS Program has about one year left to run to the end of the pilot phase although impact assessment activity may extend six months to one year beyond that.

THE FUTURE

After the pilot phase is over, the host departments may decide to go fully operational at their sites with the systems that have been developed for their departments. In the larger trials, this would mean expanding from the 100 or so workstations of the pilot phase, to as many as 2000 stations in regional offices from coast to coast. Such expansion would require financing by the host departments since there will be no further monetary involvement from the Program after the pilot phase.

We should mention that the OCS Program has a role in other pilot automation trials in the government besides the four for which it is directly responsible. Two such trials have gotten underway more recently but the funding for them has had to come from other sources within the government since the OCS funds are totally exhausted by the four trials we have described. But these two newer trials have continuing advisory input from OCS personnel. These trials are following OCS guidelines and are both benefiting from as well as adding to the Program's experience. Beyond these two newer trials, the OCS Program personnel are frequently called upon in a consulting capacity in related endeavours elsewhere in the government since they are recognized as having a considerable body of expertise in management, technology and social science.

LEARNING EXPERIENCE

It should be stressed that the entire OCS exercise is a learning experience for all concerned - vendors, host departments, the government in general, and the OCS Program itself. Although the vendors are experienced in developing software systems, only one has had previous experience developing office automation systems. For this reason, the hands-on practical exposure they will get developing and implementing systems in the field trials should be invaluable.

As is true of government and business in general, the site departments have lived through the electronic data processing movement of the '60's and 70's, but this, of course, was aimed at relatively routine, highly structured support functions. Systems intended for multi-function knowledge workers are quite new although one of the site departments has been and continues to be pioneering in exploring technology directed at such workers. This department has a number of related projects underway apart from its OCS Program trial.

Since knowledge worker systems are so new, the OCS Program is, by definition, a pioneering effort. While personal computers are beginning to proliferate throughout the government as they are in the business world, they generally fall short of the kind of integration of functions and inter-machine communications which the OCS technologies represent.

Since the trials are a learning experience, it follows that all concerned would do things in different and better ways were they now to begin similar endeavours anew. And so, even if one or another of the players decides against proceeding directly to a larger scale operational phase after the pilot phase is over, for whatever good reasons they may then have, then they will do so out of the wisdom gained in the trials. Achievement of wisdom about the new office technologies is one important justification for the entire OCS adventure for both industry and government.

IMPACT ASSESSMENT

The authors are both social scientists. We are directing the impact assessment of the four OCS trials and we also have advisory input to related government endeavours including the two newer trials mentioned above. It is still too early in the process to talk in any detail about impact assessment outcomes in the trials. The hard and soft evidence is just beginning to come in. However, partly as a result of the trials, we have learned much about procedural matters which, we feel, have crucial significance for office systems success. Before turning to these, we will briefly outline what we are trying to achieve.

ASSESSMENT OBJECTIVES

In general terms, impact assessment is an attempt to examine the events in the trials as systematically and objectively as possible, in order to learn about the intended and unintended effects of integrated office technologies on work, productivity and people. What is learned should enable some degree of future control over impacts. A second goal is to provide feedback during the course of the trials so that corrective actions might be taken when impending or actual problems are suspected or revealed. In this way, the assessment process is intended to contribute to successful implementation of the technology.

Formulation of the issues being examined was facilitated by an Impact Assessment Committee formed by the Department of Communications. The committee included representatives from various interested groups in the government.

The issues were grouped into five categories:

1. System Performance
2. Users' Acceptance
3. Human/Social Factors
4. Organizational Variables
5. Productivity

These categories encompass detailed concerns such as system usage statistics, ergonomics, user attitudes, functionality, quality of work life, health/safety/stress, morale, work transformation, social effects, training effects, demographic implications, employment effects, labour relations, organizational structural effects, work effectiveness and efficiency and system cost justification. It is intended that information on all of these categories will be useful to industry, the host departments, the government in general and the public at large.

CARRYING OUT THE ASSESSMENT

Rather than acquire the rather large staff that would have been necessary to do the detailed assessment work, the expertise of outside consultants with the necessary social and behavioural science background was sought. Another committee was involved in evaluating the responses from outside contractors to a Request For Proposals to do the work. This committee included the project director from each host site. Many excellent proposals were received. As it turned out, the best outcome, one which satisfied each project director for his own site as well as satisfied the rest of the committee for high expected quality, was obtained by hiring four contractors, a different one for each site. Spreading the work among four contractors rather than hiring a single one for all trials, has an effect analagous to that obtained by having four different system vendors; it is a better stimulus to the development of consulting expertise in the office technologies area than would have been obtained were only a single consulting firm hired. As will become clearer below, we feel that a most important element in the success of the new technologies is an appropriate understanding of offices and how people work. Social science can contribute much to this element. To now, with data processing technologies, such understanding was not as critical since the focus was on repetitive, well understood processes that could be performed by computer without a tight interaction between the person and the computer.

THE USERS' VIEW OF OPTIMAL PLANNING, IMPLEMENTATION AND ASSESSMENT

Going back now to the procedural matters alluded to previously, we will adopt the point of view of an end user, be this an individual, group or organization. This view differs in important ways from, for example, the view of a systems vendor. Planning, implementation and assessment of an office technology system can be broken down into the following five steps which are sequential except for the last one.

1. needs analysis from the users' point of view
2. baseline measures
3. system/vendor choice
4. training and learning
5. ongoing assessment

NEEDS ANALYSIS

The first step, an appropriate needs analysis is, beyond question, the first priority in any scenario of office change. All subsequent steps critically depend on and follow from the needs analysis. Errors or misconceptions at this initial stage will prove very costly to rectify later. As important as this first step is , it is the one that is least well understood. This alone may account for the bulk of failures and system rejections which preoccupy office automation professionals.

The basic problem is that an appropriate understanding of the specific, unique configuration of people, goals and events that comprise any particular office group or organization is not easy to achieve. But its difficulties do not excuse inadequate approaches. Among these, and perhaps the dominant one in use is what we will call the behaviouristic approach. Underlying this approach is something akin to a child's view of what goes on in an office. One observes fingers at keyboards, telephones being used, marks on paper, paper being circulated and being put into and taken out of metal boxes, people gathering in small and large groups in special rooms, perhaps extensive travel to get to these special rooms, and so on. With the behaviouristic approach one tries to measure the frequencies of these events by observation, counts and questionnaires. Recommendations are then made about using electronic means to accomplish high frequency events more efficiently. If the events get done faster, that is interpreted to mean a productivity increase.

Let us emphasize again, that we are dealing with the frequently unstructured work of managers and professionals and not with highly structured support functions. Although the behaviouristic approach may be quite appropriate for certain classes of events in multi-function work, we contend, a functional approach, which tries to understand the goals that are intended to be achieved both by organizations and knowledge worker individuals, is more likely to lead to the kind of analysis which will yield alignment with electronic technologies that will be viable.

The behaviouristic approach tends to lead to substitution of one means for another. A functional approach tends to lead to transformations in work. As a very simple-minded example, electronic messaging may seem to be needed from a behaviouristic perspective when two people make very frequent phone calls to each other but are often caught in telephone tag. However, if these calls are mostly about bottlenecks on a production line, then the problem is on the production line and not in telephony or messaging. More complex instances are not rare. If you have recently moved within Toronto, you will have found out that moving your telephone service is now extremely simple and fast. A single phone call to the business office will settle most matters. With his or her fingertips, the Bell representative can access all the information about whether jacks are present at your new address, and, if so, can program in the changeover date and time and can tell you your new number, right then and there. The previous long sequence of events, involving many persons, green trucks, appointments and delays, are gone. The new way did not involve more or faster trucks but a complete transformation in the way of doing things. The end goal, efficient service to subscribers, was rethought in functional, not in behaviouristic terms.

Because we don't have adequate general tools yet, reaching an understanding of organizational and individual goals may be very time consuming. It needs to involve a team, where, in addition to knowledge of the potential technical possibilities, various kinds of social science expertise is also present. Discovering the real goals of an organization requires considerable digging and analysis. A set of simple questions and observations will not do because different offices and people vary enormously in many dimensions. Workers themselves may not be able to see clearly beyond the boundaries of their present ways of doing things. Partly as a result of the difficulty in performing an adequate needs analysis, considerable uncertainty often surrounds the feasibility of an office automation system. For this reason, many organizations have opted for pilot projects before installing larger, more expensive operational systems. One of the purposes of a pilot project is to determine whether the system in fact really meets an organizational need, hence justifying the expense of a fully operational system. The OCS Program has adopted this perspective. It provides funding for the pilot phase of the project but the host department must decide whether the system should become operational and obtain the necessary funding.

It is at the needs analysis stage that the goals to be achieved by implementing technology have to be settled, as well as specification of how it will be known later, whether, or to what degree, the goals have been achieved. Much is made of the "productivity" issue, with inappropriate questions such as "how do you measure office productivity"? The real question is, "is the technology implementation (or other transformations) suggested or described by the needs analysis going to be truly useful in accomplishing the goals of this specific office or organization and its people"? If it is, then measuring later whether the technology brought about the desired changes should be relatively straightforward since detailed specification of just what changes were planned to be achieved is part of a properly executed needs analysis. The important thing here is that what is truly useful may be something quite different from office to office and organization to organization - even from person to person since knowledge workers are well-known to have idiosyncratic patterns of working. The "measuring office productivity" question is stated at the wrong level of generality and simply cannot be answered in that form. The office doesn't exist!

In this light, the cost justification issue is not, whether the new way will be cheaper or more expensive than the old. That is a substitution question. If the proposed new way is a significantly better way to achieve organizational goals, then that is what justifies it. To decide whether the costs of the new way are worth the dollars requires the usual business judgement that takes into account the same variables as any other decision on expenditure. Unlike the simple-minded substitution question, these are complex variables that can't be expressed in a simple equation.

It should be noted too that the thorough understanding of an organization, its goals and means, which a proper needs analysis will provide, does not necessarily implicate new technologies. In some instances, other means (e.g. organizational changes, job redesign, etc.) may be proposed in order to achieve improvements in goal attainments. More often than not, a combination of means, including new technologies, will be called for.

We will deal with the remaining four steps in planning, implementation and assessment, quite briefly.

BASELINE MEASURES

This is the stage where impact assessment proper begins, although the kind of expertise to do these measures resides in the team membership that performed a proper needs analysis. Unless one is a single-person organization, it is always necessary to justify major changes. Which pre-implementation measures are to be taken will follow directly from a proper needs analysis. The latter will have specified the planned effect of the new technologies and other change agents. The baseline measures are a pre-implementation series designed to be repeated post-implementation. They directly assess the effects of the events intervening between measures.

Of course, other uncontrolled and perhaps unknown events besides the implemented, planned changes can produce differences between pre and post measures. However, the incredible expense of an experiment involving a control group identically treated except for the planned changes will rarely allow this more rigorous approach to be taken. The expense includes, of course, the cost of the severe disruption to ongoing business affairs of allowing a control group. But, even ignoring the cost factor, it is almost impossible to create a group of workers that would be "identically treated except for the planned changes" since the planned changes are a great multitude of events and not some single, easily defined step. In technical terms, we are not sure what the relevant independent variables are, hence a true experiment is not feasible. In most practical situations, a case study approach is the best that can be achieved. Professional judgement will need to be exercised to interpret obtained pre and post measured changes.

SYSTEM/VENDOR CHOICE

This too should be comparatively straightforward after a proper needs analysis, since the latter would have produced a detailed specification of requirements. The competitive bid route will often be feasible. One caution here is that it may be that no vendor can meet all requirements as specified. Clear priorization of requirements may then be necessary for effective choices among vendors. Also, given that there is a choice among two or more vendors, other considerations such as the user interfaces, "user friendliness", and ergonomics should play a large role in choices.

TRAINING AND LEARNING

These issues too should have been addressed to some extent in the needs analysis phase. However, effective decisions may not be possible until after vendor choice. The vendor can frequently provide training; in other cases, in-house training staff may be used. Remembering again that we are dealing with multi-function knowledge workers, the luxury of "days" of training may not be available. The preferred route may be very brief, hours rather than days, of initial training to enable immediate, if very incomplete, usefulness of the technologies. Thence, a long term support function, preferably in-house and always on call, for gradually extending the training on both an "as needed" and a "you should next learn" basis.

We distinguish training from learning, with the latter a long-term proposition of gradually increasing mastery toward truly effective use of the new technologies in a job. Overall planning will have to anticipate the high likelihood that initially, and perhaps for a long time, individuals may become far less effective in their jobs as the jobs are transformed by the technologies. Such productivity drops are entirely normal and have to be accommodated in the often slow progress toward eventual organizational effectiveness enhancements.

ONGOING ASSESSMENT

Formally, assessment began at the second step above, with baseline measures. Ideally, assessment will be ongoing throughout the changeover, with repeated measurements being re-administered until all detectable changes cease and a new stability is attained. One constraint is not to overwhelm the workers with assessment. That will be a matter of reasonable judgement regarding both the frequency of measurement and the nature of the measures, and also a matter of establishing and maintaining rapport.

Besides providing a picture across time, repeated, ongoing assessment can play a vital role by aiding the detection of sub-optimal events.

Appropriate actions can then be taken. Such continual feedback can facilitate the total transition to the new ways of accomplishing work. Properly done, ongoing assessment becomes an integral part of the long term learning process.

CONCLUSIONS

In this paper we have described the objectives of the OCS Program and have provided a brief overview of the field trials underway. Although impact assessment results are only beginning to come in, it has become clear that an adequate needs analysis is crucial to successful office systems implementation. Currently no general techniques are available to ensure an appropriate analysis. This is unfortunate because, as we pointed out, a mistake made in the needs analysis tends to affect all of the subsequent work done on a project adversely. Indeed, it can be argued that a mistake originating in any subsequent phase of the project, can be corrected just by spending more money. However, when a mistake is made in the needs analysis, not only will much more money need to be spent, but, even then, a truly viable system may still not be obtained. This indicates the importance of conducting further research to deepen our understanding of the office and to provide a set of general tools for functional description and analysis.

THE EMERGING CORPORATE DATABASE ADMINISTRATOR: OVERVIEW

A.W. Johnston
Business Information Services
Coordinator
Shell Canada Limited
Box 400, Terminal A
Toronto, Ont. M5W 1E1

ABSTRACT

The concept of the database administration function is traced through the literature from the role of interface between applications programmers and operations staff to the coordinator of end user database production. This evolution requires an emphasis on managing the total system utilizing Database Management System software to increase productivity for department operations.

30

Une nouvelle fonction: la gestion corporative des bases de données.

Le concept de la fonction de gestion d'une base de données est recherché à travers la littérature spécialisée à partir du rôle d'interface entre des programmeurs d'applications et du personnel opérationnel jusqu'au coordonnateur de la production des bases de données. Ce cheminement requiert une emphase sur la gestion du système total utilisant le logiciel de la Base de données du Système de Gestion afin d'accroître la productivité pour les opérations du département.

31

THE EMERGING CORPORATE DATABASE ADMINISTRATOR: OVERVIEW

INTRODUCTION

The concept of a data base administration function within an organization has been discussed for the past ten years at least. According to the literature, the need for such a function was rooted in two developments. The first was the recognition that data are a valuable resource of an enterprise that have to be conserved, managed and made available for use. Secondly, as data base management systems became available, data processing departments became aware that optimizing their use required a new administrative function (Meltzer 1975).

As Herbert Meltzer remarks in his overview of the administration of data bases, during the first generation of data administration, the economics of computer usage appeared to require the most efficient use of hardware at the expense of the most efficient use of people. Some enterprises may not yet have evolved from that era. However, further development brought the recognition that stored data had to be shared among applications. Among the difficulties that arise are problems of economics, the inability to respond to each of the sharers in a timely, cost-effective manner, problems of discipline, the interference of the sharers with each other, and the problems of change - the inability to evolve the database without time consuming and costly modifications to large numbers of programmes and large quantities of stored data (Meltzer, 1975). To solve these problems and act as an interface between applications programmers and the computer systems operations people, many organizations set up a separate administrative function.

END USER PROGRAMMING

As the various departments in a company have attempted to use the power of the computer to aid their productivity, management have responded with a number of different ways of structuring the resources that provide assistance in the development of computer systems. Data processing professionals continue to be a scarce and expensive resource. The literature was filled with articles suggesting solutions to long applications queues. In 1974 in Don Mills, IBM formed the Information Centre to teach non data processing employees how to access the computer for their own needs. As this particular Information Centre concept has spread to many companies throughout North America it has influenced the evolution of the data processing department. The prediction is that the 1980's will see the data processing department become the manager of the data resource, just as the finance department manages the cash resource. Finance creates a chart of accounts and develops and enforces policies for managing cash. Data processing will be asked to develop a data architecture upon which systems can be designed and built and they will develop and enforce policy and procedures for managing data (EDP Analyzer 1981).

No doubt the proliferation of information centres and software packages has led many departments to assist themselves to manage their information needs using the computer. From the point of view of the whole enterprise this led to questions of redundancy, accessibility and security. Many of us also remember how ephemeral many of those systems were. They were often poorly documented and once the developer moved on to a new company or the programming support staff went on to other projects, the system faltered and finally failed.

Concurrently we began to see articles in journals such as the *Internal Auditor* and the *Journal of the Society for Management Accountants* that were concerned with standards and security for the data a company stored in its computer. These articles introduced the terminology "data base" and "data base management systems" to their readers. They looked upon the data base as a collection of inter-related data stored together on a direct access storage device such as a disk with controlled redundancy to serve more than one application. They called for a standard and consistent approach to add new or modify old data items to the data base (Lobley 1981). The point of view of these articles is not the usefulness of the data to the work of the end user but with the security, data validating procedures and reliability of the data being stored. With multiple users accessing a company's data from a variety of viewpoints they wanted to know who was in charge. They see the data base administrator acting like the controller in protecting the interests of the organization. (Anderson and Techavichit 1982).

DATABASE MANAGEMENT SYSTEMS

As we progressed through the 70s, the problems of standards, redundancy and end user satisfaction were to be met by our beleagured data processing departments using data base management system software. These systems, such as INQUIRE and FOCUS, are a collection of software required for using a database, and presenting multiple different views of the data to the users and programmers (Martin 1982). In addition to all that it was already doing, the data processing organization was being called upon to coordinate systems development on a product by product basis rather than on a system by system basis. For example, with the use of INQUIRE the industrial hygiene department could automate the update of their Materials Safety Data Sheets while research could track developments of the substances involved. The reports from these two data bases under INQUIRE would not look at all alike and could be maintained in TSO, CICS or a number of other operating environments.

It was still the case that the data processor usually considered only the automated portion of the system being developed as being "the system". Little concern was given to the total user environment and how the computer portion fitted into the total system design from the department's point of view (Weiss 1981). In such a situation, even when a data base administrator was appointed, the role was often restricted to compiling data dictionaries and standardizing file access methods. In the meantime, the software vendors emphasized easy-to-use report writers and online query features of their products. In this environment the DBA must function more as a coordinator and less as a technician (Computer World 1982).

In 1982 James Martin defined a data base administrator as an individual with an overview of one or more data bases who controls the design and use of these data bases. He goes on to say that it is often better to use two individuals, the data administrator and a data base designer who designs the physical aspects of the data base. In this setup the data administrator is the individual with the overview of an organization's data, responsible for designing the data model and obtaining agreement about the definitions of data which are maintained in the data dictionary. I haven't met such an individual and I haven't seen an organization which has set up its data model. In the meantime, it has been widely recognized that one individual could scarcely hope to carry out all these functions in a large organization so we start seeing articles about the data administration group whose functions included: planning and user coordination, standards and documentation, software acquisition, database design and performance management (Datamation 1984).

It is not surprising that Datamation had another article in their January 1984 issue which was titled "Wanted: Experienced Kamakazee Pilots" (Sweet 1984). As the data base technology grew so did the call for more data base administrators. Data Management in 1983 did a survey of starting salaries and produced a reading list of the latest data base books. Where do data base administrators come from? Most of the technical training seems to come from on-the-job training and vendor courses. A recruiter remarked that programming was probably the most useful place to begin a career in data base management.

END USERS AND DATA BASE MANAGEMENT SYSTEMS

From the viewpoint of those of us in departments attempting to use computerized information to carry out the objectives of the company, there are problems with data base administration being confined to the data processing world view. Data base administrators were being called upon to elicit their customers needs much as application programmers had, as well as to develop tools such as the data dictionary and to fit their data bases into the operating systems of their companies. For a description of an alternative source of expertise in these areas, look at Datamation September 1983 for a readers' forum article "Librarians: The Untapped Resource" (Koenig 1983).

To look beyond the computerized section of a total information system in an environment requires an information specialist with much wider background than programming code. One corporation described what they were looking for as follows: an information specialist with a specific subject background in the subject matter of the data base; a knowledge of the source of the material accessed by the data base whether it be a physical filing system or parts of other data sets or data bases; a knowledge of commercial data base services and other sources of information for the enterprise. Such a data administrator would be trained on the particular software being used by his company. The function would coordinate the user departments' requirements for reports and maintenance routines with the data processing department's standards for the best system features to meet those needs. While the data dictionary development may still be lodged in the data processing department, there is a requirement for another kind of standard for data base input, i.e. standards for the actual data that are going in the fields which have been defined by the data dictionary.

In many of our organizations, data entry continues to be the low level position which keypunch was, but online systems very often do not have the verification procedure which was built into keypunch shops. The data administrator was therefore seen as the person who would conduct the training for the actual data entry clerks and would monitor the output of the systems. To explore the emerging corporate data base administration role from the user department standpoint, I introduce Kamakaze pilot James Barker.

REFERENCES

ANDERSON, E.H. and TECHAVICHIT, J.V. "Data-base Systems and the Controller", in The Internal Auditor, February 1982, pp. 46-49.

FRANCELLA, K. "The Data Base Administrator Comes of Age", in Data Management, February 1983, pp. 34-35.

KOENIG, M.E.D. "Librarians: The Untapped Resource", in Datamation, September, 1983, pp. 243-244.

LOBLEY, D. "Data Bases and the Role of the Data Dictionary", in Cost and Management, March-April 1981, pp. 42-44.

MARTIN, J. Strategic Data-Planning Methodologies. Englewood Cliffs, N.J., Prentice-Hall, 1982.

McNURLIN, B.C. "Supporting End User Programming", in EDP Analyzer, June 1981, pp. 1-12.

MELTZER, H.S. "An Overview of the Administration of Data Bases", in American Federation of Information Processing Societies; Information Processing Society of Japan. 2nd USA-Japan Computer Conference Proceedings, August 26-28, 1975, Tokyo, Japan, pp. 365-370.

PROIA, A. "DBA's Role Changing as Package Use Rises", in Computerworld, November 15, 1982, pp. 39-40.

SHAH, A.D. "Data Administration: It's Crucial", in Datamation, January 1984, pp. 187-192.

SWEET, F. "Wanted: Experienced Kamikaze Pilots", in Datamation, January 1984, pp. 197-198.

WEISS, H.M. "Making the DBA the Key to System Development", in Canadian Datasystems, March 1981, pp. 66-70.

THE EMERGING DATABASE ADMINISTRATOR

J.L. Barker
Database Administrator
Records Management Department
Shell Canada Limited
505 University Avenue
Toronto, Ontario M5G 1X4

ABSTRACT

The Database Administrator (DBA) is a direct product of the introduction of text database management system technology into modern corporate administration. As a powerful and versatile administrative tool, the main objective of the database management system is to store and retrieve information through and in the form of natural business language. The power and flexibility of such systems, however, are also the causes of system failure at the implementation stage. Database management systems deal with natural language, and language represents human concepts. They are also designed for end user defined applications. As both concepts and applications fluctuate in any business environment when departmental needs change, it requires human management to ensure that such system changes follow suit. The ultimate need for highly qualified professional human management of database management systems has led in recent years to the emergence of the database administrator as a middleman between user and programmer. Though traditionally resident in the Data Centre, the DBA is increasingly emerging as a member of a user department. Consequently, he is in a strategic position to carry out a role that involves all facets of database management from database design to report construction. This paper therefore outlines his role from the viewpoint that a DBA resident in the user department is more advantageous than one remotely located in a Data Centre, and that his main purpose is to make the system as transparent to his departmental users as possible.

Une nouvelle base de données de gestion.

La Base de Données de Gestion (BDG) est un produit direct de l'introduction de la technologie du système de base des données de gestion à texte à l'intérieur d'un conseil d'administration moderne. Comme outil administratif puissant et versatile, l'objectif principal du système de base de données de gestion est l'entreposage et le recouvrement de l'information à travers et sous la forme d'un langage des affaires naturel. La puissance et la flexibilité de tels systèmes, néanmoins, sont aussi la cause d'une défaillance du système au stade de l'implantation. Les systèmes de base de données de gestion opèrent avec un langage naturel, et le langage représente des concepts humains. Ils sont aussi dessinés pour des applications bien définies. Comme les concepts et les applications varient en même temps dans n'importe quel environnement des affaires quand le département a besoin de changements, il demande à la gestion «humaine» de s'assurer que de tels systèmes changent dans la même direction. Le besoin ultime pour du personnel de gestion hautement qualifié dans les systèmes de bases de données de gestion a mené, durant les dernières années, à la naissance du gestionnaire de bases de données qui sert de tampon entre l'utilisateur et le programmeur. Par les résidents habituels du Centre de Données, la BDG apparaît de plus en plus comme faisant partie du département de l'utilisateur. En conséquence, il occupe une position stratégique afin de jouer le rôle qui implique toutes les facettes d'une base de données de gestion, de la conception de la base de données jusqu'à la composition du rapport. Cette communication de ce fait souligne son rôle dans le sens qu'un résident de la BDG dans le département de l'utilisateur est plus avantagé que quelqu'un, au loin, localisé dans le Centre de Données, et que son utilité principale est de rendre le système aussi invisible que possible aux utilisateurs de son département.

In many large corporations, management continues to wrestle with the introduction of computer technology into the office environment as a means of easing the burden of manual office work, improving internal communications and paper management. This technology is defined generally under the umbrella of office automation. The wrestling match however continues as the actual implementation is often confounded by five major pitfalls:

(a) Vendor claims of what their systems can do are often misleading;

(b) Systems fail to achieve intended objectives;

(c) Cost benefits of implementation fail to materialize;

(d) Systems frequently overlap;

(e) Adequate professional or high level support where the technology is being introduced is absent.

These symptoms of incomplete or inadequate implementation, especially the lack of professional support, are particularly evident when corporations make the momentous decision to introduce, as part of the office automation package, text database management systems (DBMS).

The common denominator of most office technologies is simply the management of information, frequently in the form of paper documentation. Whatever systems are chosen to perform this function, few are actually designed to store and retrieve large quantities of information over long periods consisting of several years. The exception is the large mainframe-oriented database management system, a sophisticated software package designed to store and retrieve information through and in the form of natural business language. The power and versatility of such software to meet the applications requirements and specifications of end users on one hand and to use other mainframe software and utilities through the Programme Language Interface on the other cannot be denied. Their power and flexibility however are at the very heart of the problems experienced during implementation and the ongoing operation of a DBMS.

The causes of a DBMS to falter can be reduced to two fundamental factors:

(a) management's serious underestimation of the requirements for supporting such a system;

(b) the need to deal with natural language.

Language represents human concepts. As both concepts and the applications designed to handle them fluctuate in any business environment whenever departmental needs change, it also requires human management to ensure that the system changes accordingly. The administration and professional level of support needed to ensure the smooth implementation and continued operation of such a software package are the areas most often neglected by senior management whenever database management systems are introduced as part of office technology. This support consists of close and carefully coordinated co-operation among three areas:

(a) Systems support from the Information Center;

(b) Computer Services support for purposes of ensuring adequate mainframe resources to run the software;

(c) End user administration of the individual applications.

In effect, office automation is living proof that invention is often the mother of necessity, because the need for end user participation in a sophisticated text handling environment has given rise to a very critical need for skilled support staff within the user department. Management frequently ignores this area of administration, equating information handling with low level clerical support functions. Regretably this attitude leads to disaster. The success of a particular application within a user department is solely dependent on database design. In other words, it is technique not technology that is most important. Consequently, as text database management systems are developed and become more sophisticated, the need for proper database management on a high professional level has become essential in order to achieve system stability within the user department and make the system work for the user instead of vice versa. Recognition of this fact has slowly led to the emergence of the Database Administrator.

Although relatively new to the office environment, the qualifications and profession status of a Database Administrator are growing almost in direct proportion as the power and level of sophistication of database management systems increase. The Database Administrator is therefore a direct product of installing text DBMS technology into the administrative framework of any modern corporation. His emergence is directly linked to the nature of the software involved.

The single overriding objective of database management software is to provide end users with an efficient and effective administrative tool that will meet their information handling needs by permitting the user department to define its own database or application. By exposing the end user increasingly to user friendly software, the system decreases the need for constant professional programmer intervention, although it does not entirely eliminate that intervention. This trend towards end user nonprogrammer computing has necessitated a greater knowledge within the user department of the software's user language and maintenance facilities than is generally possessed by the average clerical support staff. The role of the Database Administrator is therefore that of middleman, between end users on the one hand and the system support people and programmers on the other. Though not usually a programmer by profession, the Database Administrator possesses three essential assets:

(a) He has a thorough understanding of his own department's information needs;

(b) He has an equally thorough knowledge of the DBMS capabilities and how to use them effectively;

(c) He understands the needs of Systems Support staff with respect to such areas as database analysis and definition.

With an indepth knowledge of his own department's operational, and hence information handling needs, the DBA is able to effectively perform several vital tasks. First and foremost is database design, that is determining what information is to be stored within a database and how it is to be retrieved from that database. In the case of the INQUIRE installation in Shell Canada, the Systems Support staff has written several CLISTS to help a DBA perform this function. The DATABASE CLIST, for example, is used to set up a test database for purposes of running different types of reports and analyzing the data. Before a test database can be submitted to Systems Support to be put into production, the Database Administrator must determine, through an anaylsis of the field structure, which fields are to be used for retrieval purposes, the optimum length of each field and the field type.

Changes in departmental information needs may also necessiate a change in database design. Periodic statistical analysis of a database is carried out by Systems Support and given to the Database Administrator. It is the responsibility of the DBA to use the analytical data to determine if a particular application or database requires a partial or complete revision of its field structure. Generally the analysis indicates the degree of usage of a particular field and how it is being used. With this data, the Database Administrator can approach the end users within his department to inform them of what changes are required and why they are necessary. The objective is to streamline a database to improve retrieval and simplify database maintenance.

Knowledge of the database management system's capabilities is equally essential for the Database Administrator to perform three further functions. First the DBA seeks to optimize report formats through the more advanced parts of the system's User Language. In the case of INQUIRE, a further CLIST called the INQUIREB CLIST is used to construct large batch reports both in test and production databases. In addition, the Automatic Macro facility can be effectively used to design on-line and batch reports interactively.

The second function arising from a thorough knowledge of the sytem's capabilities is the Database Administrator's ability to select the optimum method of maintaining a database. Again in the case of INQUIRE, the DBA has the choice of three possible approaches, depending on the type of maintenance to be performed. The selection is dependent on the Database Administrator's knowledge of which method is the most efficient. These methods consist of:

(a) a menu-driven approach through the use of Interactive Macros, generally written by the Database Administrator in the system's User Language;

(b) the Full Screen Processor (FSP) which is an interactive screen editing function;

(c) Global changes, again using the system's User Language.

Menu-driven maintenance is achieved by storing several Interactive Macros that are supported by the Data Management Supervisor (DMS). These macros are constructed in User Language by the DBA with assistance from Systems Support. Their function is to provide a sequence of interactive prompts for purposes of correcting data online.

With respect to the FSP, Systems Support is responsible for creating a set of screens, called a Panel, for each database wherever the user group wishes to use this facility. The DBA's role in this instance is to provide Systems Support with the initial Panel design based on a sound understanding of a database's field structure.

The global change function is one of the most powerful and time-saving capabilities of the User Language at the DBA's disposal. This facility allows the DBA, with one query, to search the entire database for the field value to be changed and write the commands to replace the data into a dataset called a SAVE FILE. After checking the SAVE FILE to ensure that the replacement commands are correct, it is then submitted through the INQUIREB CLIST as a batch maintenance job. A recent example involved switching the names of refineries in a database called MATIS from one field to another and deleting the names from their original field. Consequently, the DBA can perform in minutes a task that normally takes days to complete.

The final task performed by the Database Administrator involves optimizing the storage of data within a database. The result can mean dramatic savings in computer resources with respect to the amount of storage space taken up by the information. Available to the Database Administrator for this purpose are two utilities, namely the FDT CLIST and the DATABASE CLIST. Both utilities permit construction of such space-saving devices as Flat Files and manipulation of the field structure through the User Language.

Hence, a DBA can determine in test mode which field structure is most efficient for a particular application before giving the database to Systems Support to be put into full production. Weeks of development time can therefore be saved during the transition of an appplication from test to production mode.

In conclusion, the role of DBA has emerged out of the need for human management of text database management systems within end user departments. That role covers a wide range of activities that are primarily intended to bring stability to the end user as far as getting the most value out of a very complex software. Essentially, the DBA attempts to mask the complexities of the software and render it as user-friendly and transparent to his own end user group as possible.

Database Generation by End-Users

Raymond G. Blackie
Analyst Programmer
Shell Canada Limited
Toronto, Ontario
M5G 1X4

Abstract

Many Database Management Systems (DBMS) set formidable barriers to the development of database applications by non-technical end-users. Users are forced to turn to programmers to have them build the databases they need. This results in a bottleneck that effectively discourages vital user participation during application development. This paper outlines techniques that have been successfully used with INQUIRE, a text DBMS, to provide end-users with the tools they need to build databases that they may use in a fashion consistent with the production environment.

La création dynamique de base de données par les utilisateurs

Certains Systèmes de Gestion de Bases de Données (SGBD) élèvent des barrières insurmontables dans le développement d'applications de bases de données par des utilisateurs à formation non technique. Ces utilisateurs sont alors obligés de consulter des programmeurs afin de construire les bases de données dont ils ont besoin. Cela aboutit souvent à une impasse qui décourage efficacement la participation vitale des utilisateurs durant le développement des applications. Cette présentation souligne les techniques qui ont été utilisées avec succès avec INQUIRE, un SGBD à textes, pour fournir aux utilisateurs les outils dont ils ont besoin pour construire des bases de données qu'ils peuvent utiliser d'une manière compatible avec la production de l'environnement.

Database Generation by End-Users

INTRODUCTION

Most Database Management Systems are so complicated that a programmer handles the creation of the databases. The complexity arises in the attempt to be all things to all people. This in turn creates a bottleneck - the programmer - who must be consulted before you can have your database.

This applies to INQUIRE, a text DBMS, that has been used at Shell Canada for the past three years.

To establish a production INQUIRE database it is necessary to amass sixteen (16) pieces of information. These include the database name, sample data, the Field Definition Table, average record size, capacity, key length, direct access structure characteristics and space allocations for the component files.

When you put all of this information together, you end up with a database that has a direct access structure and a static Field Definition Table. The database may be searched, either sequentially or via the direct access structure, or maintained by multiple concurrent users.

However, such a database must be frequently backed up and periodically reorganized. All maintenance transactions should be logged. A recovery procedure for damaged databases must be available. User access to the system must be monitored for chargeback purposes.

To add insult to injury, the process of establishing such a database often takes weeks to complete. This includes discussing the Field Definition Table with the user, discussing standards, selecting keyed fields, appointment of a Database Administrator, paperwork, determination of the database characteristics and insertion of the database into the production system.

Few non-technical users are willing to wait and endure a rigorous analysis of their application. The Information Centre concept has shown that, given appropriate tools, users can develop their own applications. The user wants to enter data, build a database and query the database - today!

FIELD DEFINITION TABLE

The Field Definition Table, or FDT, describes the fields in the database. The description includes the name, format, length, print format and type of each field.

Field formats include fixed length text, variable length text, unpacked decimal, binary, short floating point and long floating point. The ones most relevant to our discussion are the fixed and variable length text formats and unpacked decimal. The other formats are special numeric formats that could not be entered by a user at a terminal.

Field types include scalar, repeating, subfield, element or group. Since subfields, elements and groups are merely reinterpretations of previously defined fields and do not themselves require any space in the database records, they may be excluded from our discussion. They may be added later if necessary. This leaves us scalar of repeating fields.

Scalar fields may be either unpacked decimal or fixed or variable length text fields.

Repeating fields have either a fixed or variable number of repeats. They may be either unpacked decimal or fixed or variable length text. The one restriction is that a repeating field may not be variable length text and have a variable number of repeats.

The format and type of field determine the amount of space required to hold the field.

ISI

Infodata Standard Input (ISI) format is the format in which the data is expected to be in when building an INQUIRE database.

Traditional flat files put field values in specific columns and the space for a field is reserved whether it is used or not. The length of a field must be known in advance so that columns can be assigned. This is a "horizontal" view of the data.

ISI format, however, is a "vertical" approach. In its simplest form, the field name is entered in columns 1 to 8 and the data in columns 10 to 72. Long fields may be continued on subsequent lines. Fields that are empty need not appear. Records are delineated with an 'END' statement.

ISI format promotes fielding of data and allows the data to use whatever room it requires. Fields are not tied to specific columns and there is no need for fancy line continuation schemes.

Data entry is aided by an ISI command that provides a full-screen form that the user fills in with field names and data.

SIMPLIFYING THE PROCESS

The key to providing an appropriate tool for end-user database creation is the realization that the two most difficult aspects of database creation are the determination of nine (9) characteristic numbers associated with the direct access structure and the construction of a suitable Field Definition Table for the database. Users are also notoriously poor at estimating the amount of space they need for a database.

The first step is to sacrifice the direct access structure and this has several important ramifications. It reduces the number of parameters necessary to define a database from sixteen to seven. All queries must sequentially search the database. The database cannot be maintained directly - instead, you must change your data and rebuild the database. The database does not need backup, reorganization, logging of maintenance, recovery procedures or access monitoring.

The second step is to create a Field Definition Table for the user. This requires that the data be entered in ISI format. The ISI data is analyzed to pick up the field names and various field characteristics.

The final step is to determine the disk space required to hold the database. The database usually requires less than 50% of the space required to hold the ISI data and that information is readily obtained from the system catalogue.

Now we have reduced database creation to two (2) parameters: a database name and data in ISI format. No further reduction is possible.

ANALYZING THE ISI DATA

Historically, this was the first part of the process to be developed. It was originally intended to pinpoint inefficiently stored fields. Part of its function was to propose an alternative FDT that would minimize the storage necessary for each field. It would also identify unused or under-utilized fields.

The first step in the analysis involves gathering information on each field in the ISI data for each record. We need to know the field name, the total length of data over all repeats in the field, number of repeats, and length of the longest repeat. Is the field numeric? If so, how may decimal places does it have? is the field empty?

From this, we can continue on to establish some overall characteristics of each field. We are now in a position to calculate the maximum number of repeats, the longest length of a field value and the average number of repeats. We now also know how many records we are dealing with. We can also answer our earlier questions regarding blank and numeric fields. If the field is numeric, the field length must be adjusted to minimize the loss of decimal places.

The final step is to decide which of the various possible field configurations actually minimizes storage for each field. Once the format and type have been decided for a field, an entry can be generated for the FDT.

FDT entries consist of a field name, format, length, print format, type and possibly the number of repeats.

GENERATING THE SEQUENTIAL DATABASE

As outlined in the introduction, it is only necessary for the user to supply 2 pieces of information in order to build a sequential database.

In the system, as I have set it up, the user would type: DATABASE PROJ to create a sequential database named PROJ. The user would then be prompted for the name of the dataset containing the ISI data. Once this information has been entered, a batch job is created and submitted.

This is what the user sees - let us briefly look at what goes on behind the scenes.

After the user enters the ISI dataset name, the system catalogue is checked for the size of that dataset. The size of the database is then estimated at less than 50% of that size. (If it turns out that the database actually requires more room than the estimated size, additional smaller chunks of space will be provided.)

The batch job includes steps to delete any currently existing database under that name, delete any currently existing FDT under that name, analyze the ISI data and generate an FDT, validate the FDT and load the ISI data into the sequential database. (The database and FDT generated are specific to one user so multiple users may use the same database names.)

This system could be run online but I have chosen to run it in batch because the reports that it produces may be too long to be displayed on a screen.

QUERYING THE SEQUENTIAL DATABASES

Production databases are accessed through the commands INQUIRE (online) and INQUIREB (batch). Users may generate reports, control certain parameters, create datasets on disk or tape, print multiple copies of a report, use special forms, print reports in simplex or

48

duplex and perform maintenance on their databases through the query language.

All of these capabilities, except for the performance of maintenance, exist for the sequential databases. This is an extremely important point! Users should not have to learn a multitude of systems just to get at their database. The path from development to production should be consistent and similar. The only additional information that the user need provide is the fact he is working with a test sequential database.

At the command level, production databases use either direct access or sequential access to retrieve records. Sequential databases are restricted to sequential access (and hence the name).

Since the maintenance commands of the query language rely on direct access, sequential databases cannot be maintained through the query language.

APPLICATIONS

Applications for sequential databases break down into three general classes: personal, design and review.

Sequential databases are not particularly useful for long-term applications or large applications where many people need fast access to some of the records. Applications that require even a moderate amount of maintenance may be unsuitable.

Personal applications include calendaring, project management and a phone book among others. These applications tend to be highly individualistic and involve very few records.

Designing new applications becomes much simpler because the user can do the development on his own. The user may gain new insights into his application before putting it into production thus saving himself and the programming staff a lot of work in the long run.

Reviewing applications is important. Over the lifetime of a database, fields may need to be added, others will no longer be used or the nature of the data changes. Users are prompt to ask for new fields but are often unaware that certain fields are unused or are could be structured more efficiently. The report generated by the analysis of the ISI data provides sufficient information for investigating deletion or alternative storage configurations for each field. To review a production application, it must be dumped into a dataset in ISI format.

EXTENSIONS

The most obvious extension is to have the DATABASE command accept an FDT from the user. If the user provides the keyword NOGEN after the database name, the user will be prompted for the dataset name containing the FDT. The batch job will not delete the current FDT and it will not analyze the ISI data.

This leads to the second extension - verification of the FDT. If users are to develop FDTs, they should be able to validate them without the necessity of loading records into a database. If the keyword NOGO is supplied after the database name, the FDT will be validated and no database will be created.

The third extension is to allow users to specify the amount of space for the database. The parameter TRKS(17) will provide 17 tracks for the database.

The final extension is to permit the loading of flat files into sequential databases. Flat file formats are impossible to analyze without knowledge of the field layout. Since the user knows the field layout he must provide the FDT. Since the user is providing the FDT, analysis of the data is bypassed. The only restriction with flat files is that there may only be one (1) variable type field in the FDT. The DATABASE command will still estimate the database size but it uses a different formula because flat files are assumed to be denser in data than ISI files.

SUMMARY

The system, as I have designed it, consists of five (5) pieces.

First, the user requires a method for entering the data. The ISI command provides a screen on which the user enters field names and data.

Secondly, the user may wish to create his own FDT. Screens are provided for this purpose in the FDT command.

The key component is the DATABASE command as it brings together the ISI data and an FDT to form a sequential database. The FDT is often generated by the DATABASE command through analysis of the ISI data. The DATABASE command also often determines the space necessary for the sequential database.

The final two (2) components provide the query environment for the sequential databases. These are the commands INQUIRE and INQUIREB. Any differences or restrictions pertaining to the sequential databases are due to the nature of sequential databases and are not artificially imposed. The lessons the user learns here are applicable to databases in the production environment.

Sequential databases as I have described them should be viewed as augmenting the production environment to provide a smooth and consistent path into production for end user applications.

THE IMPACT OF FIFTH-GENERATION COMPUTERS ON LIBRARY AND INFORMATION SERVICES

MICHAEL A. SHEPHERD
SCHOOL OF COMPUTER SCIENCE
TECHNICAL UNIVERSITY OF NOVA SCOTIA
HALIFAX, NOVA SCOTIA, CANADA B3J 2X4

ABSTRACT

Over the past forty years, many predictions have been made of computer systems which would be "intelligent"; able to organize data, infer solutions, and make decisions without human intervention. It appears that the fifth-generation of computers may finally fulfill some of those predictions. In anticipation of such intelligent systems, librarians and information specialists should not ask, "What will fifth-generation systems do for library and information services?", rather, "What do we want library and information services to be by the time fifth-generation systems are readily available?" This paper gives an overview of fifth-generation systems and their potential impact on library and information services.

L'impact des ordinateurs de cinquième génération sur les bibliothèques et les centres de documentation.

Depuis quarante ans, on a fait de nombreuses prédictions au sujet des systèmes d'ordinateur qui seraient «intelligents», capables d'organiser les données, de proposer des solutions et de prendre des décisions sans aucune intervention humaine.

Dans l'attente de ces systèmes intelligents, les responsables des bibliothèques et centres de documentation ne devraient pas se demander: «Qu'est-ce que les ordinateurs de cinquième génération apporteront aux bibliothèques et aux centres de documentation?», mais plutôt: «Que désirons-nous que soient devenus ces bibliothèques et centres de documentation lorsque les systèmes de cinquième génération seront disponibles?»

Ce document donne un aperçu des systèmes de cinquième génération et de leur impact possible sur les bibliothèques et centres de documentation.

THE IMPACT OF FIFTH-GENERATION COMPUTERS ON LIBRARY AND INFORMATION SERVICES

MICHAEL A. SHEPHERD
SCHOOL OF COMPUTER SCIENCE
TECHNICAL UNIVERSITY OF WESTERN ONTARIO
HALIFAX, NOVA SCOTIA, CANADA B3J 2X4

INTRODUCTION

Almost forty years ago, Vannevar Bush recognized that the basic processes of information retrieval could be reduced to machine processes and the effect that this might have on the storage and retrieval of information (Bush, 1945). Twenty years later, Licklider (1965) characterized the computer-based library of the future as being able to:

- handle both documents and facts;
- represent a body of knowledge and improve on that representation through use;
- use heuristics to solve problems;
- automatically improve its performance through learning techniques such as feedback;
- network users for both knowledge input and retrieval;
- negotiate the user query.

Although current systems exhibit some of the above characteristics, there are no systems which have all of the above capabilities. Fifth-generation computers, however, will be "intelligent" and should be able to provide the above capabilities and possibly more.

These intelligent systems are no longer merely the wishful thinking of a few farsighted individuals. The time frame for delivery of fifth-generation systems is the mid 1990's although commercial products incorporating various features of these sytems will be in the market place by 1985 (Schefter, 1983). As the potential impact of such systems on library and information services is profound, planning for their use should begin now, not after the systems are already available.

FIFTH-GENERATION COMPUTERS

CHARACTERISTICS OF FIFTH-GENERATION SYSTEMS

The characteristics these machines will exhibit are best exemplified by the three broad targets of the Japanese Fifth-Generation Computer Systems project: the improvement of man/machine communication, the development of systems able to infer solutions to problems, and the storage, organization, and retrieval of nonnumerical information (McCorduck, 1983; Richmond, 1982).

It is expected that man/machine communication in fifth-generation systems will be through continuous speech. These machines should be able to understand continuous human speech with a vocabulary of 50,000 words from a few hundred speakers with an accuracy of 95 per cent.

Typical of the fifth-generation inference systems will be expert systems which should not only be able to translate fluently between languages, but should be able to automatically organize the contents of documents into knowledge bases. Expert systems will make use of these knowledge bases to provide consultative services to both subject specialists and to the general public.

The third broad target of the Japanese project is a system capable of the organization and retrieval of nonnumerical information as required for picture and image processing. Picture and image processing systems require the capability to deal effectively with the characteristics of three dimensional spaces, satellite images, and medical images.

The goal for the fifth-generation computer projects is to produce an inference computer or computer network capable of 1 billion logical inferences per second with a knowledge base of a billion inference rules and a database of hundreds of million of objects. In order to achieve this goal it will require a blending of three areas of science and technology; VLSI, computer architecture, and artificial intelligence.

FIFTH-GENERATION COMPUTERS

FIFTH-GENERATION COMPUTER MEMORY

Much as the personal computer has been given to us by large-scale integration of computer circuits and the supercomputers by very large-scale integration, ultra large-scale integration will play an important part in the delivery of the fifth-generation. These silicon chip-based components provide the internal memory and are the realization of the computer architectures that run the software.

The memory capacity of a silicon chip can be measured in bytes or characters. In order to supply the fifth-generation with the projected memory capacity, a million chips will have to be linked together, each chip with a capacity of 128 kilobytes, for a total of 128 billion bytes of internal memory.

The memory capacity of the silicon chip has doubled every eighteen months since 1974, with the 32 kilobyte chip being developed in 1983 (Pease, 1983). In keeping with this timetable, the 64 kilobyte chip should be developed in mid-1985 and the 128 kilobyte chip the end of 1986. However, in October 1983, IBM announced the production of an experimental 64 kilobyte chip, somewhat accelerating the timetable (IBM Corp., 1983).

In order to put a prototype 128 billion byte computer into perspective, it should be noted that the Encyclopedia Britannica has approximately 200 million characters.

FIFTH-GENERATION COMPUTER ARCHITECTURE

The architecture of current and past generations of computers may be described as the von Neumann architecture, after the mathematician who conceived it. The von Neumann architecture has a global addressable memory to hold program and data objects, with program instructions frequently updating the contents of the memory during execution. It also has an instruction counter which holds the address of the next instruction to be executed. This counter is either implicitly or explicitly updated to provide the machine with a sequence of instructions to execute.

The sequential nature of this architecture has become a limiting factor in the development of faster machines even though the physical components of the machines are faster than ever before. Each instruction to be executed must be fetched from the memory by the CPU along with the operands specified by the instruction. All results from the execution of the instruction must be stored in memory before the next sequential instruction, specified by the instruction counter, is fetched for execution.

In order to break this sequential bottleneck, a variety of parallel architectures have been proposed. In a parallel architecture, instead of one instruction being executed at a time, a number of instructions may be executed at the same time with the results then being used for other parallel operations. Machines based on parallel architectures promise to be orders of magnitude faster than von Neumann architecture machines.

ARTIFICIAL INTELLIGENCE

One of the major areas of research in artificial intelligence (AI) has been the development of "intelligent" machines. An intelligent machine is a computer program that can perform cognitive tasks at high levels of performance, not necessarily mimicing the internal structure of human behaviour (Feigenbaum, 1983).

The field of artificial intelligence will provide the intelligence aspects of fifth-generation computers in the form of expert systems. An expert system is a computer program whose performance is intended to rival that of human experts (Duda and Shortliffe, 1983). Such systems are knowledge-based in that they depend more on the presence of a large body of knowledge than on ingenious computational processes.

There are a number of such expert systems in the market place running on the present generation of computers. These systems are limited by lack of internal memory, lack of processing speed, and lack of a universal knowledge base. The lack of memory and processing speed should be rectified by the fifth-generation computers, after which it will be feasible to build larger, more complex knowledge bases.

Expert systems usually consist of two major components, the knowledge base with its underlying inference structure, and the database. By separating the two it is possible to enlarge the database or change attributes of the database without having to change the knowledge base.

The knowledge and inference structure may be represented in the form of rules, normally of the IF...THEN nature. An example of a rule from the MYCIN system, used in the selection of antibiotics for patients with severe infections, is given in figure 1. Assuming that there is suggestive evidence of pseudomanas aeruginosa as in the example, this knowledge would then be used as the predicate for another IF...THEN rule which may suggest a treatment.

```
IF the infection is meningitis
   AND organisms were not seen
       in the stain of the culture
   AND the type of infection
       may be bacterial
   AND the patient has been seriously burned
THEN there is suggestive evidence that
     pseudomonas aeruginosa is one of the
     organisms that might be causing
     the infection.
```

FIGURE 1: EXAMPLE OF MYCIN RULE

The knowledge in the MYCIN system is matched against the database, in this instance the data on a particular patient. Patient data are stored as <attribute-object-value> triples, as shown in figure 2.

The stain (ATTRIBUTE) of a particular
organism (OBJECT) is gram-negative (VALUE).

FIGURE 2: EXAMPLE OF MYCIN DATA

CURRENT LIBRARY AND INFORMATION SYSTEMS

Most current library and information systems can be depicted as having a centralized database accessed from two separate sides; the administrative side and the user side. This is particularly true of library systems which, for the most part, have been designed by librarians for librarians. The user side of such systems have far less scope than the administrative.

On the administrative side, the term, "library automation", has a close analogy in, "factory automation". One of the major functions of a library automation system is inventory control with its cataloguing, circulation, and ordering subsystems.

The user side of such a system allows the user to query the database and retrieve citations. The user cannot update the inventory directly, order new materials, place reserves on items, or view his own circulation history.

FUTURE SYSTEMS

Future systems, based on fifth-generation computers, should allow a merging of the administrative and the user systems. The administrator may access the system for different purposes than the user, but it will be the same system.

The database itself will be stored on computer, not just the references to the database. This database may be distributed across the country but will be accessible anywhere there is an interface and the fact that it is distributed will be transparent to the user.

The interface, whether for the administrator or the user, will be a natural language interface. The dialog may be carried out verbally or through a terminal. The system will remember past sessions and be able use this knowledge to provide a user-friendly interface and refined search strategies. The system will operate at a level indicated by past sessions and by the ongoing query negotiation. In this way, a naive first-time user will have a system dialog tailored to his level.

The knowledge in the database and the underlying inference structure itself will be represented. The user will be able to retrieve documents, facts, and concepts. The system will also be able to analyze, store, and retrieve nonnumerical data such as graphics, still pictures, and moving pictures. The user will be able to ask the system to infer solutions to problems whose answers are not already in the database.

The database will be tied to electronic publishing and updated automatically from the contents of the input data.

The knowledge organization itself will change automatically depending on changes in the database and in patterns of use.

KNOWLEDGE ENGINEERING

If the fifth-generation computer projects deliver systems with capabilities fulfilling the stated goals, there will be a profound change in the manner in which we access and use data.

The very fact that these systems are "intelligent" will require them to be well managed. Procedures regarding database maintenance and access will have to be established. Access procedures determine who has access to what knowledge base as well as what system has access to what data and to what knowledge base. Maintenance procedures determine the vetting of information before it is added to the database and the development and vetting of rules incorporating the knowledge prior to adding the rules to the knowledge base.

Smith (1980) has referenced Tou (1979) who defines knowledge engineering as that discipline that deals with planning, design, construction, and management of knowledge-based systems for transfer, utilization, and extension of knowledge. The goal of a knowledge engineer is to make knowledge more generally accessible; something which librarians have been doing for some time. The main difference is that a new and very powerful tool will soon be at our disposal.

CONCLUSIONS

Fifth-generation systems will provide at least the characteristics of Licklider's library of the future. These systems will be able to:

- handle documents, facts, concepts, and pictures;
- represent the body of knowledge and the underlying inference structures and automatically update both the representation and the data;
- use expert systems to solve problems;
- automatically improve its performance through the use of long-term memory (past sessions) and short-term memory (feedback);
- network users for both knowledge input and retrieval;
- negotiate the user query in natural language, either through voice communication or terminals.

With such a powerful tool soon to be at our disposal, the question is not, "What will fifth-generation systems do for library and information services?", but rather, "What do we want library and information services to be by the time fifth-generation systems are readily available?"

FIFTH-GENERATION COMPUTERS

REFERENCES

BUSH, V. "As We May Think." *Atlantic Monthly*. 176(1). (July 1945): 101-108.

DUDA, Richard O., and SHORTLIFFE, Edward H. "Expert Systems Research." *Science* 220 (April 1983): 261-268.

FEIGENBAUM, Edward A. "Artificial Intelligence." *IEEE Spectrum* 20 (November 1983): 77-78.

IBM Corp. "Experimental 512k Chip." *Data Product News* 5 (October/November 1983): 11.

LICKLIDER, J.C.R. *Libraries of the Future*. Cambridge, MA: MIT Press; 1965.

McCORDUCK, Pamela. "Introduction to the Fifth Generation." *Communications of the ACM* 26 (September 1983): 629-630.

PEASE, R.F.W., "Fabrication Issues for Next-Generation Circuits." *IEEE Spectrum* 20 (November 1983): 102-105.

RICHMOND, Louis S. "Here Comes Computer Inc." *Fortune* 106 (October 1982): 82-84.

SCHEFTER, Jim. "Fifth-Generation Computers." *Popular Science* 222 (April 1983): 79-81.

SMITH, Linda C. "Artificial Intelligence Applications in Information Systems". *Annual Review of Information Science and Technology*. 15 (1980): 67-105.

TOU, Julius T. "Knowledge Engineering". Unpublished Paper available from the author, Center for Information Research, University of Florida, Gainsville, FL 32611. (1979).

Centre de Toxicologie du Québec
Le Centre Hospitalier de lUniversité Laval
2705, boul. Laurier
Québec, Qué

"Gestdoc" ou l'informatisation de la gestion de documents à l'intérieur d'un centre de documentation spécialisé.

BENEDETTI, J.-L.; LEFEBVRE, L.; SANFAÇON, G.; WEBER, J.-P.; GALLICHAND, S.

Résumé

Créé en 1971 par le Ministère des Affaires sociales, le Centre de Toxicologie du Québec (C.T.Q.) s'est vu attribuer en 1974 la responsabilité du réseau provincial de lutte contre les empoisonnements. En 1975, il obtient du ministère un premier mandat en toxicologie industrielle, mais l'attribution de nouveaux mandats en 1979 a amené le Centre à une restructuration interne par la création d'une section Information-Documentation. Cette section offre ses services principalement au réseau des Affaires sociales (salles d'urgence, centres antipoisons, départements et centres locaux de santé communautaire) mais aussi aux autres ministères et organismes divers. Les domaines couverts sont la toxicologie médicamenteuse et générale et la toxicologie industrielle et environnementale. Afin de répondre aux demandes qui lui sont adressées, la section a, à sa disposition, un centre de documentation spécialisé en toxicologie comprenant des volumes de base ou spécifiques, des périodiques et une importante collection d'articles de revues touchant les divers domaines couverts par le Centre. Pour faciliter la tâche aux divers professionnels du C.T.Q. affectés à la réponse de ces demandes, il a été décidé d'informatiser la gestion des tirés-à-part (TAP). Après enquête auprès de plusieurs bibliothèques et centres de documentation, le C.T.Q. a opté pour un système de classification qui lui est propre. Tous les programmes informatiques utilisés dans la gestion des TAP sont codifiés en langage APL et entreposés sur un des ordinateurs de l'Université Laval.

Le premier groupe de programmes a trait à la commande des TAP, enregistre toutes les données relatives au TAP, fait l'impression sur différentes formules selon que la demande est faite directement à l'auteur ou à une bibliothèque. Le deuxième groupe consiste à la réception des TAP, à la vérification des données entrées précédemment et à la transmission au demandeur. Le troisième groupe a trait à l'indexation des TAP. Le demandeur, à l'aide d'un thésaurus, attribue au TAP un descripteur principal ainsi qu'une section principale et une sous-section qui sont vérifiés et/ou modifiés par le responsable de l'indexation finale. Les divers programmes de ce groupe enregistrent toutes données et font l'impression d'étiquettes et de cartes permettant le classement ordonné du TAP. Le quatrième groupe de programme permet de faire des recherches informatisées des TAP comme nous pouvons le faire avec des systèmes, tels que Medlars, Dialog, etc.

Pour terminer, divers autres programmes complètent l'informatisation des TAP et nous permettent de suivre l'évolution du système.

«Gestdoc», or the computerization of document management within a specialized documentation centre.

Abstract

Set up in 1971 by the Ministry of Social Affairs, the Quebec Toxicology Centre (C.T.Q.) has been given, in 1974, the responsability of the Provincial Anti-Poisoning Network. In 1975, the Ministry gives the C.T.Q. a first mandate in industrial toxicology, but the assigning of new mandates in 1979 has led the Centre to an internal reorganization by the creation of an Information/Documentation section. This Section offers its services mainly to the Social Affairs network (emergency rooms, antipoison centres, health departments and local centres), but also to other various ministries and organizations. Fields covered are medicinal and general toxicology, and industrial and environmental toxicology. In order to answer requests addressed to it, the Section has at its disposal a Documentation Centre specialized in toxicology and comprising basic or specific reference books, periodicals and an important collection of magazine articles about the various fields covered by the Centre. To simplify the work of the various C.T.Q. professionals assigned for answering these requests, it has been decided to computerize the management of offprints. After investigation in several libraries and information services, the C.T.Q. has opted for its own classification system. All computerized programs used in the offprint management are coded in APL language and stored in one of the computers of Laval University.

To conclude, several other programs complete the offprint computerization and help us follow the development of the system.

"GESTDOC" OU L'INFORMATISATION DE LA GESTION DE DOCUMENTS À L'INTÉRIEUR D'UN CENTRE DE DOCUMENTATION SPÉCIALISÉ

"Gestdoc" ou l'informatisation de la gestion de documents à l'intérieur d'un centre de documentation spécialisé.
BENEDETTI, J.-L., LEFEBVRE L., SANFAÇON G., WEBER J.-P., GALLICHAND S.

INTRODUCTION

Le Centre de Toxicologie du Québec est responsable depuis 1974 du programme provincial de lutte contre les empoisonnements. En 1979, l'attribution de nouveaux mandats en toxicologie industrielle a nécessité une restructuration interne du centre, concrétisée entre autres, par la création d'une section information-documentation.

Ce nouveau mandat, de même que l'implication de plus en plus fréquente du centre dans le domaine de l'environnement a amené une croissance extrêmement rapide de la documentation nécessaire pour répondre aux divers intervenants du réseau des Affaires sociales, des ministères et organismes divers, du public et des professionnels du C.T.Q.

1. ÉVALUATION DES BESOINS

En 1979, le C.T.Q. possédait une collection d'environ 10,000 tirés-à-part, lesquels étaient simplement rangés par ordre alphabétique de sujet, selon l'idée principale retenue par l'utilisateur. Aucun système de gestion de documents n'avait alors été mis en place.

La gestion de cette collection représentait le problème principal du C.T.Q.

Une enquête auprès de plusieurs centres de documentation spécialisés et de bibliothèques permit d'inventorier les différents systèmes de classifications et classements utilisés, les différentes conceptions et les différentes fonctions documentaires. Nous avons alors constaté que la plupart des centres de documentation qui ont des collections spéciales (ex.: films, tirés-à-part) utilisent une classification et un classement qui leur sont propres, le classement de ces collections s'intégrant difficilement dans les grandes classifications telles la classification de la Library of Congress ou la classification décimale Dewey.

Il semble donc que le système "maison" présente plusieurs avantages dont celui de classer les tirés-à-part selon les sujets les plus traités au Centre, selon ses conceptions précises et surtout en fonction du besoin des utilisateurs.

De plus, la classification "maison" permet au Centre de modifier sa propre classification selon l'évolution de ses besoins. Cependant, il en résulte une incompatibilité complète entre les différents organismes traitant de sujets identiques ou voisins.

2. OBJECTIFS

Après l'évaluation des besoins des utilisateurs du C.T.Q., nous avons donc opté pour un système de classification "maison".

Le système choisi doit donc permettre de simplifier la commande et la réception des documents, de classifier les documents de façon à permettre un accès direct et rapide à tous les documents de la collection.

3. GESTDOC

"Gestdoc" est cependant plus qu'un système de classification. C'est un système informatisé de gestion de documents. Il permet, outre l'indexation et le classement, la commande des tirés-à-part, la préparation des fichiers et étiquettes ainsi que la recherche bibliographique manuelle et automatisée. Il régit aussi la circulation des tirés-à-part à l'intérieur du centre et le prêt aux utilisateurs externes.

"Gestdoc" est effectif depuis 1980.

Le système GESTDOC est subdivisé en groupes (ou sous-programmes) qui correspondent aux différentes fonctions effectuées par le programme.

3.1 Commande des tirés-à-part.

Le groupe GENTRE permet l'entrée des références bibliographiques en vue de la commande des tirés-à-part.

La commande des tirés-à-part est faite directement à l'ordinateur et imprimée sur les formulaires conçus pour les demandes aux auteurs ou aux bibliothèques (groupe GIMPRIME).

Le groupe GENTRE permet aussi la saisie des références des tirés-à-part possédés par le C.T.Q. avant 1980 et qui doivent être indexés pour faire partie de la banque de données du C.T.Q.

Ce programme vérifie si la référence est déjà dans la banque évitant ainsi le dédoublement des commandes.

68

3.2 Réception des tirés-à-part

Le groupe REÇUTAP permet l'attribution d'une date d'arrivée au document et confirme au niveau de la banque de données que le document existe dans la collection du C.T.Q.

L'exactitude de la référence est alors vérifiée et les corrections nécessaires sont apportées à l'aide du groupe GCORRIGE.

3.3 Traitement des tirés-à-part

3.3.1 La classification

Section principale:

Les préoccupations du C.T.Q. étant axées vers plusieurs domaines connexes, mais très différents, il a été convenu d'établir des sections principales (SP) correspondantes à chacun de ses domaines d'intérêts:

TCG - Toxicologie clinique et générale
TIE - Toxicologie industrielle et environnement
DOC - Documentation et gestion

Descripteur principal:

Un descripteur principal est attribué à chaque document à l'aide du thésaurus élaboré au C.T.Q. Le classement se fait par ordre alphabétique du descripteur principal.

Sous-section:

La sous-section (SS) correspond à une subdivision thématique à l'intérieur des dossiers. Un certain nombre de sous-sections ont été choisies en fonction des principaux aspects traités pour chaque dossier au C.T.Q.

Ex.: AN - Analyse
TA - Toxicologie aiguë

Renvoi:

Lorsque plus d'un concept est nécessaire pour la recherche manuelle d'un document, un renvoi à un descripteur secondaire est effectué. Ce système n'est pas automatisé et non exhaustif, puisque la recherche de documents spécifiques peut se faire à l'aide de banques de données commerciales

(Medlars, Dialog, etc.). Les fichiers "auteurs" permettent de localiser le document au Centre.

Code de localisation:

Ce code est attribué à chaque tiré-à-part et inscrit sur le document et sur les fiches correspondantes. Il est formé des abréviations des SP et SS, du descripteur principal et du numéro du document:

Ex.: SP / DP / SS / No
T I E / MERCURE / AN / 123

3.3.2 **L'indexation**

Les normes d'indexation ont été définies en fonction des besoins du C.T.Q., à l'aide des normes de l'Association française de normalisation (AFNOR).

Chaque tiré-à-part est d'abord indexé par le demandeur qui soumet à l'indexeur ses propositions concernant la SP, le DP et la SS, ainsi que les renvois.

L'indexation provisoire se fait à l'aide d'un bordereau transitoire d'indexation.

71

L'indexation définitive sera faite par l'indexeur qui, à l'aide du thésaurus du C.T.Q. et autres outils documentaires, transpose les concepts retenus en langage contrôlé.

Les données relatives à l'indexation sont enregistrées sur le système informatisé à l'aide des programmes du groupe **GMOTCLÉ**.

Le groupe **GMOTCLÉ** permet ensuite l'impression des étiquettes et cartes des divers fichiers.

3.3.3 Classement:

Le classement se fait selon la classification et des étiquettes de couleurs permettent d'identifier la SP.

Ex.: T C G - rouge
T I E - bleu
D O C - orange

Chaque dossier est identifié par le descripteur principal et subdivisé selon les SS identifiées par deux lettres.

3.4 Élagage de la collection

La pertinence de conserver un document est établie par les professionnels du Centre en fonction de la langue du document, du sujet, de la qualité de l'article et de l'âge du document.

Le programme permet d'indiquer qu'un document a été détruit, mais la référence est conservée dans l'ordinateur afin d'éviter la duplication de la commande de documents.

3.5 Recherches bibliographiques

Le système de gestion "GESTDOC" permet, en plus de la commande de l'indexation et du classement des tirés-à-part, la recherche manuelle ou automatisée des documents répertoriées dans le système.

La recherche manuelle, plus économique et souvent plus rapide (panne de l'ordinateur, etc.), se fait à l'aide du thésaurus et de deux fichiers, soit numérique ou chronologique d'entrée et ordre alphabétique d'auteur.

La recherche automatisée se fait en mode conversationnel. Le système est une version très simplifiée des

73

banques de données commerciales. On peut chercher les tirés-à-part par mot du titre, descripteur et auteur. La logique booléenne permet des interactions.

4. PROGRAMMES COMPLÉMENTAIRES

D'autres groupes de programmes permettent de dresser la liste des périodiques et de leur code, de connaître la date de réception d'un tiré-à-part, de dresser la liste des tirés-à-part reçus pendant une période donnée, de connaître le pourcentage des tirés-à-part reçus.

5. AVANTAGES ET INCONVÉNIENTS DU SYSTÈME

5.1 Inconvénients

Le système demande beaucoup d'efforts afin d'assurer la mise en place. Les ressources humaines nécessaires pour assurer l'indexation des tirés-à-part reçus avant 1980 sont insuffisantes. Par conséquent, il est impossible d'avoir un inventaire exhaustif de la collection.

Le rodage des programmes informatiques est très long et des améliorations sont toujours nécessaires.

Les coûts pour la recherche bibliographique automatisée rendent son utilisation difficile.

La mise en place d'un langage contrôlé et la préparation du thésaurus est très difficile à cause de l'étendue des domaines d'application. De plus, les équivalences entre termes anglais-français sont souvent difficiles à établir.

5.2 Avantages

GESTDOC permet d'éliminer la duplication des commandes de tirés-à-part.

La classification permet de retrouver en peu de temps les documents recherchés.

Les tâches techniques, telles préparation des fichiers, étiquettes, etc. sont non seulement raccourcies, mais, de plus, la présentation est améliorée et plus uniforme.

À l'exception du groupe "RECHERCHE", le programme est peu dispendieux et l'utilisation d'APL sur l'ordinateur de l'Université Laval assure un accès continu aux fichiers à moins de panne importante.

L'utilisation d'un langage contrôlé (thésaurus) permet de rassembler sous un même terme toute la documentation portant sur un sujet et simplifie l'accès à l'information pour les professionnels du Centre pendant une période donnée et de faire imprimer les références des tirés-à-part non reçus, etc.

CONCLUSION

GESTDOC est donc un système de gestion de documents encore en évolution. Des étapes de développement sont prévues afin de répondre à l'avancement du C.T.Q.

On pourrait ainsi envisager d'étendre le programme aux autres types de documents, ex.: volumes, rapports.

D'autre part, l'indexation pourrait être plus poussée afin d'inclure les descripteurs secondaires dans la banque de données.

APPLYING UNIVERSAL RELATION THEORY TO BIBLIOGRAPHIC DATA

R.G. Crawford
Associate Professor
Department of Computing & Information Science
Queen.s University
Kingston Ontario
Canada

ABSTRACT

It is appropriate that a bibliographic retrieval system should be designed based on a coherent data model. In particular, the relational view of data is easily applied to bibliographic data. However, some aspects of this model that are advantageous for systems design present difficulties for the user. In particular, the advantages of normalized relations may be lost on the user who must navigate an apparent maze of relations to express a query. Universal relation theory offers a potential solution to this difficulty. Examples using a universal relation in the Ingres relational database system are discussed.

Théorie de la relation universelle appliquée à la conception bibliographique.

Il est essentiel qu'un système de recherche bibliographique soit conçu sur la base d'un modèle cohérent. En particulier, l'interrelation des données est facilement applicable à la conception bibliographique. Cependant, certains aspects de ce modèle qui sont à l'avantage du design des systèmes présentent des difficultés pour l'utilisateur. En particulier, les avantages de relations normalisées peuvent confondre l'utilisateur qui doit alors naviguer dans un labyrinthe apparent de relations pour exprimer une demande. La théorie de la relation universelle offre une solution potentielle à cette difficulté. Des exemples utilisant une relation universelle dans le système de banque de données relationnelle de Ingres sont discutés.

UNIVERSAL RELATION

INTRODUCTION

The advantages of the relational view of data for bibliographic data have been previously described (Crawford 1981). The relational model provides a simple, coherent, consistent, and generally natural approach in this area. The design of a normalized set of relations for bibliographic data is conceptually easy. Nevetheless, there are obstacles to the use of such a database system. While it may be possible to express very powerful queries, some of the more mundane queries may be somewhat cumbersome.

In the following section, this problem is shown explicitly in the INGRES relational database system. The next section introduces universal relation theory, and this is followed by a section in which the application of this theory is demonstrated.

BIBLIOGRAPHIC RETRIVAL USING INGRES

INGRES, a relational database management system, has been used for bibliographic retrieval. Some results have been reported previously (Crawford 1983). One of the conclusions of that work was that it is necessary to permit use of a bibliographic retrieval system in a way that users find natural, and that such an approach was not apparently facilitated by INGRES.

This current report focuses on the use of views in INGRES to facilitate retrieval of bibliographic data. In particular, viewing the data as a universal relation is considered.

INGRES (Interactive Graphics and Retrieval System) (Stonebraker et al 1976) (Stonebraker 1980) is a relational database system that is implemented on top of the UNIX operating system. INGRES is written in C and runs in UNIX as a user job.

Bibliograhic Database

The database consists of five normalized relations. These are listed below with the relation name followed by a list of attributes in parentheses. The names are largely self-explanatory; docno (document number) is a unique integer value assigned to each document.

CITATION (docno, title, journal, month, year)
AUTHOR (docno, name)
KEYS (docno, keyword)
CATEGORIES (docno, category)
ABSTRACT (docno, text)

A careful description of normalization as applied to bibliographic data may be found in (Crawford et al 1983a)

Retrieval in INGRES

The data manipulation and query language of INGRES is QUEL. QUEL is a tuple relational calculus language. This means that range statements and range variables are an important part of the language. It is non-procedural and includes a comprehensive set of arithmetic and aggregation operators.

For the QUEL examples sown in this paper we assume the following range statements.

```
RANGE of C is CITATION
RANGE of A is AUTHOR
RANGE of K is KEYS
RANGE of T is CATEGORIES
RANGE of B is ABSTRACT
```

As an example of retrieval using QUEL, consider the following approach to retrieving title and year for all documents having "compiler" listed as a keyword.

```
RETRIEVE (C.title, C.year)
WHERE C.docno = K.docno
AND K.keyword = "compiler"
```

This query is a fairly clear expression of what is desired. However, notice that it is necessary to specify the join (C.docno = K.docno) of the CITATION and KEYS relations.

Consider an even more complicated example.. Again, retrieve all references to "compiler", but in this case retrieve more information, including title, author, journal, year and abstract. The expression of this query in QUEL is:

```
RETRIEVE (C.title, A.author, C.journal, C.year, B.abstract)
WHERE C.docno = A.docno
AND   C.docno = B.docno
AND   C.docno = K.docno
AND   K.keyword = "compiler"
```

It is unacceptable to require a user to express in such a complex way what is a fairly simple request. Users generally have in view an entire document, and are not interested in knowing that the document.s author, title, and keywords are in three different relations, each of which must be named. An approach to overcoming this particular problem may be found in universal relation theory.

UNIVERSAL RELATION THEORY

The concept of a universal relation for a relational database has been a subject of discussion recently (Ullman 1982) (Maier & Ullman 1983) (Kent 1981). A universal relation is a single relation whose schema consists of all the attributes in any of the relational schemas of the database. (Assume attributes representing the same thing in different relations are given the same name and attributes representing different things are given different names.) The purpose of a universal relation is to provide the database user with a simplified model in which he can compose quieries without regard to the underlying structure of the relations in the database.

UNIVERSAL RELATION

The universal relation is not a physical storage structure for the data in the database. If data were stored in a single relation, anomalies could arise when inserting into, deleting from or updating the database. Although the user sees and queries a single relation representing the entire database, the data is in fact being stored in a number of normalized relations. The user is not required to know what the attributes and schemas are for each of the normalized relations. The user need not worry about how to join relations to answer queries. It is up to the database management system to interpret and handle the queries by ensuring that the appropriate relations are joined for a specific query.

A bibliographic database lends itself well to the concept of a universal relation. In many instances, a naive user is querying a bibliographic database to obtain some or all information regarding a particular document, author or subject (category). With a universal relation the user is able to formulate queries based on any of the attributes very easily. The user need not know how joins are made on normalized relations or understand in great detail how queries are interpreted.

In order to present the user with a single universal relation containing all the information stored in the underlying normalized relations, it may be necessary to pad out some of the tuples in the universal relation with null values.

A formal description of a universal relation for bibliographic data is given in (Crawford et al 1983b). Suffice it to say here that, for each of the theoretical problems associated with the universal relation (Kent 1981), either the problem does not apply to our data, or a reasonably simple solution can be found. Here our concern is the practical implementation of the universal relation concept in INGRES.

The Universal Relation in INGRES

INGRES permits the definition of views on the relations. This provides several capabilities. For example, as a privacy provision, a user could be permitted a view of a certain relation that did not include all attributes of the relation. Or, for convenience, a user may wish to view the database in ways that involve joining two or more relations to construct the view.

For our database, the universal relation is that view that involves the join of the 5 constituent relations; and therefore includes the following attributes:

UNIVERSAL (udocno, utitle, ujournal, umonth, uyear,
uname, utext, ucategory, ukeyword)

UNIVERSAL RELATION

The appropriate INGRES command for defining the universal relation is:

```
define view universal (udocno = C.docno,
       utitle = C.title, ujournal = C.journal,
       umonth = C.month, uyear = C.year,
       uname = A.name, utext = B.text,
       ucategory = T.category, ukeyword = K.keyword)
       where B.docno = A.docno
       and   C.docno = A.docno
       and   T.docno = A.docno
       and   K.docno = A.docno
```

Now, queries can be stated in terms of this one relation. We must define a range variable on this relation, such as:

```
RANGE of U is UNIVERSAL
```

Consider our previous two examples of queries, restated in terms of the universal relation. The first is:

```
RETRIEVE (U.utitle, U.uyear)
WHERE U.ukeyword = "compiler"
```

The join term is omitted because it already is encompassd by the view.

The second query takes the form:

```
RETRIEVE (U.utitle, U.uauthor, U.ujournal, U.uyear, U.uabstract)
WHERE U.Keyword = "compiler"
```

This further emphasizes the facility with which queries on the Universal relation may be expressed. Note that we are using attribute names with the prefix "u" here to distinguish them from the attribute snares in the original relations. This is not necessary, and in fact we would not want our universal relation attribute names to be so unweildy.

CONCLUSIONS

For bibliographic data, the advantages of the universal relation are apparent. The user can compose queries without regard to the underlying structure of the component relations. Further, none of the difficulties which may attend, in theory, to the universal relation, pertain to its use for bibliographic data. This emphasizes the naturalness of the relational view for bibliographic data.

There are still problems attendant to the display of results that are stored relationally. However, these may be handled by writing additional software to format the output in bibliographic applications.

UNIVERSAL RELATION

REFERENCES

CRAWFORD, R.G., "Bibliographic Retrieval Using a Relational Database System", in Proceedings of CAIS Annual Conference, (May 1983), 108-116.

CRAWFORD, R.G., "The Relational Model in Information Retrieval", in Journal of the American Soceity for Information Science, Vol. 32, No. 1 (January 1981), pp. 51-64.

CRAWFORD, R.G., BECKER, H.S., OGILVIE, J.E., "A Relational Bibliographic Database", Technical Report #83-149, Department of Computing & Information Science, Queen.s University, September 1983.

CRAWFORD, R.G., BECKER, H.S., OGILVIE, J.E., "Universal Relation Theory As Applied to Bibliographic Data", Technical Report #83-150, Department of Computing & Information Science, Queen.s University, (1983), 13 pages.

KENT, W., "Consequences of Assuming a Universal Relation", ACM Transactions on Database Systems. 6(4) : 539-556; 1981.

MAIER, D., ULLMAN, J.D., "Maximal Objects and the Semantics of Universal Relation Databases", ACM Transactions on Database Systems. 8(1) : 1-14; 1983.

STONEBRAKER, M. "Retrospection on a Database System", in ACM Transactions on Database Systems Vol. 5, No. 2 (June 1980), pp. 225-240.

STONEBRAKER, M., WONG, E., KREPS, P., HELD, G., "The Design and Implementation of INGRES", in ACM Transactions on Database Systems, Vol. 1, No. 3 (September 1976), pp. 189-122.

ULLMAN, J.D., "Principles of Database Systems", Rockville, Maryland : Computer Science Press Inc. : 1982.

Documents as Arrays

Ian A. Macleod,
Dept. of Computing
and Information Science,
Queen's University,
Kingston,
Ontario.

ABSTRACT

There has been a significant amount of interest recently in the integration of document retrieval with data base management. Most of this work has centred on the relational model. However there are significant defects with the relational approach. In particular, normalisation causes problems. Documents typically contain repeating fields such as author names, keywords, citations and so on. Normalisation causes what was once a logical unit of information to be dispersed across a number of relations. This greatly complicates the retrieval process. Other problems with the relational model include a lack of hierarchy and the inability to retrieve objects of different types "simultaneously".

In this paper we outline the array model. Here objects are represented as array members rather than set members. Further array members may themselves be arrays. This type of information structure seems to more accurately correspond to the structure of documents. We look at the implications of such a model with respect to the query language and we also look at implementation considerations.

Représentation matricielle des documents dans les bases de données relationnelles.

On remarque beaucoup d'intérêt, depuis quelque temps, pour l'intégration du repérage des documents avec la gestion des bases de données. La plupart des études effectuées sur le sujet ont utilisé le modèle relationnel. Il y a toutefois plusieurs défauts importants à l'approche relationnelle, notamment des problèmes de normalisation. De façon générale, les documents contiennent des champs répétitifs, tels les noms d'auteurs, les descripteurs, les références bibliographiques, etc. Le processus de normalisation fait que ces unités logiques d'information sont dispersées dans un certain nombre de relations différentes, ce qui complique considérablement le processus de repérage. Le manque de hiérarchie et la difficulté de repérer simultanément des objets de nature différente sont d'autres problèmes liés au modèle relationnel.

Le présent exposé introduit le modèle matriciel dans lequel les objets sont représentés comme des éléments d'une matrice plutôt que des éléments d'un ensemble. Il est même possible que certains éléments d'une matrice soient eux-mêmes des matrices. Cette façon de structurer l'information semble mieux correspondre à la structure même des documents. L'auteur analyse enfin les diverses implications du modèle sur le langage d'interrogation et aborde quelques considérations pratiques de fonctionnement.

INTRODUCTION

By and large, information retrieval systems are primitive examples of information systems. They have predetermined search strategies, often can only handle one data base at any one time and are generally inflexible. With the evergrowing availability of machine readable bibliographic data and the increasing accessibility of inexpensive hardware, there is a corresponding growth in interest in flexible document retrieval systems.

Recent work in this area has suggested the adaption of existing data base models to document retrieval. Most of these ideas have focussed upon the relational model, [1,2,3,4]. Of the three "traditional" data base models, this has seemed the most attractive mainly because of the completeness of the associated high level query languages such as SQL, [5], as well as a certain naturalness with the underlying tabular data structures. This naturalness derives from the observation that many of the structures in information retrieval map into tables.

At first sight the relational model seems well suited to this type of application since a lot of the underlying information structures in a document retrieval system are tabular - things like dictionaries, stop word lists, indexes, thesauri and so on, and when we view the logical data organisation as a set of tables, then the relational concepts are ideal. However, when we try to extend the application of the model to more structured data, a number of the deficiencies of the relational model became more apparent.

THREE APPROACHES TO RETRIEVAL

In this section, we illustrate the relational approach together with that taken by a more conventional retrieval system, STAIRS, [6], and look at some sample dialogues. We will compare these approaches with our proposed new model, the array model.

STAIRS is a widely available commercial product distributed by IBM. It is an example of a "conventional" retrieval system produced to handle bibliographic data. It is also the system on top of which BRS, [7], a MEDLINE derivative, has been developed. STAIRS permits a document to be made up from up to eight formatted (fixed length), fields and one variable length text field. The ability also exists to partition the text field into named paragraphs. Two retrieval commands are provided. The first, SELECT mode, performs retrieval based on the content of the formatted fields. The second type of retrieval, SEARCH mode, is based on the words contained in the unformatted text field. The results of both retrieval commands can be combined using the usual set operations.

In SELECT mode, we can search for a particular value of a formatted field or a range search can be carried out. In SEARCH mode, searching is based on the presence of a particular word. Further, words can be required to be in a particular paragraph; a boolean combination may be requested; two words may be required to be adjacent or present in the same sentence or paragraph. Various weighting algorithms can be applied to the results of SEARCH mode operations if something other than strictly boolean retrieval is desired.

For example, suppose we had a document consisting of a title, a set of one or more authors, an abstract a set of keywords and a year of publication. The conventional approach here would be to have a single formatted field called "YEAR" and to split the text field into three "paragraphs", (the term is slightly misleading), called, perhaps, TITLE, AUTHORS, KEYS and ABSTRACT.

In STAIRS, this information would roughly be stored as shown in Figure I. The nested box denotes a repeating field.

```
DICTIONARY

+----+-------+
|word|pointer|
+----+-------+
      |
      |
      |        INVERTED FILE
+---------+----------+-----------------------+
|doc_count|word_count|+-----+----+----+------+|
|         |          ||docno|para|sent|wordno||
|         |          |+--| --+----+----+------+|
+---------+----------+---|--------------------+
                         |
       ------------------
       |                    DOCUMENT
+-----------+----------------+          +----+
|doc_pointer|formatted_fields|          |text|
+-----------+----------------+          +----+
     |                                     |
     ---------------------------------------
```

FIGURE I - Approximate STAIRS File Organisation

In the relational model, this information would most appropriately be spread over three relations. These might be

DOCUMENT (D#, Title, Abstract, Year);
AUTHORS (Author, D#)
KEYS (Key, D#)

This organisation is illustrated in Figure II.

```
DOCUMENT
+--+-----+--------+----+
|D#|Title|Abstract|Year|
+--+-----+--------+----+
|                      |
|   DOCUMENT Tuples    |
|                      |
+----------------------+


AUTHORS          KEYS
+------+--+      +---+--+
|Author|D#|      |Key|D#|
+------+--+      +---+--+
| AUTHORS |      | KEYS |
| Tuples  |      |Tuples|
+---------+      +------+
```

FIGURE II - Corresponding Relational Organisation

The approach presented here is to suggest a new model, the array model. This is a somewhat radical step but, as will be noted, the model is based upon a generalisation of the relational model, rather than being totally new. Array theory has grown out of attempts to generalise APL, particularly to extend APL to permit non-rectangular arrays. These are arrays where elements of the same array may have different dimensions and types. Trenchard More at IBM has been one of the leading proponents of array theory, [8,9]. The main interest from our point of view is that the basic information structure seems to be a more reasonable model of reality than do the sets of the relational model.

In the array model, the example document we had earlier can be represented as a single array as shown in Figure III.

```
+-----+---------+--------+------+----+
|Title|+-------+|Abstract|+----+|Year|
|     ||Authors||        ||Keys||    |
|     |+-------+|        |+----+|    |
+-----+---------+--------+------+----+
```

FIGURE III - Corresponding Array Organisation

Each array element consists of an array of five items - a title, an array of authors, an abstract, an array of keywords and a year. Because non-atomic attributes are allowed there are no normalisation problems. The result of retrieving something from this array is itself an array. The array model is really a generalisation of the relational model. Instead of retrieving tables from tables we retrieve trees from trees.

At this level significant differences between the three approaches are already noticeable. In STAIRS, formatted (non-text) fields are treated quite differently from text fields. Also there is a limit, though not an unduly restrictive one, on the number of formatted fields and paragraphs that are allowed. In the relational model, repeating fields such as keys and authors are treated differently from non-repeating fields, such as the title. This is caused by the normalisation process where relations are designed so as to optimise updating. (For a more detailed discussion of normalisation problems see, for example, [2]). This creation of several relations from the original document necessitates the introduction of a new field which we have called D#. Its purpose is to relate together the entries in the different tables which were derived from the same original document. As we shall see, it is this normalisation process that causes many of the problems with the relational model. In the array model, the internal organisation parallels the external view of the document.

We will now look at some queries expressed in the notation of the three systems systems and compare them. There are a number of relational query languages, all with similar power. We will use SQL, probably the best known, to demonstrate the relational approach. To illustrate the array model, the query language we use is AQL. This is an experimental language currently under development. It combines the basic structure of SQL with a subset of the operations of NIAL, a programming language based on More´s work, [10]. More detailed descriptions of AQL are given elsewhere, [11,12].

In the following examples, the STAIRS query is given first followed by the same query expressed in SQL and AQL.

a) Find all documents dated "1984".

```
SELECT
   YEAR EQ 1984

SELECT D# FROM Document
   WHERE YEAR = 1984

SELECT FROM Documents
   WHERE Year = 1984
```

The first difference to note is that in SQL we specify what fields we want retrieved as part of the search command. In STAIRS this decision is postponed until the actual display or printing of the documents. In SQL, if we want to alter the displayed information, the search must be performed again. The AQL query is identical to the SQL query except for the minor difference that if the fields to be retrieved are not specified, the entire document is retrieved. More significantly, the array model does not prohibit indirection. In particular, it is possible to obtain a list of references (or pointers), to documents, rather than the actual documents themselves. For example, we can write:

```
SELECT REFERENCE FROM Documents
   WHERE Year = 1984
```

This type of retrieval more closely follows what happens in STAIRS.

In STAIRS, the results of the retrieval operation are numbered automatically and it is possible to refer to the results using this number in subsequent operations. In both SQL and AQL, the user can assign names to results as in:

```
   ASSIGN Myname: SELECT etc. (SQL)

   Myname GETS SELECT etc. (AQL)
```

b) Find any document whose title is "WAR AND PEACE".

In STAIRS we can do a search and then restrict the search against title paragraphs as follows:

```
   SEARCH
      1: "WAR" ADJ "AND" ADJ "PEACE"
      2: 1.TITLE

   SELECT D# FROM DOCUMENT
      WHERE Title = "WAR AND PEACE"

   SELECT FROM Documents
      WHERE Title = "War and Peace"
```

This query cannot really be specified in STAIRS without making "Title" into a formatted field. If it is an unformatted paragraph, the normal choice, the best we can do is search for the adjacency of the words of the title. In SQL and AQL, this query is structurally identical to the previous one. However in STAIRS we can search on the individual words of the title whereas in SQL and AQL it would first be necessary to build a separate structure to hold the words.

c) Find all documents containing the keyword "INFORMATION".

In STAIRS we would write:

```
   SEARCH
      KEYS.INFORMATION
```

In SQL, the corresponding query is:

```
   SELECT D# FROM Documents, Keys
      WHERE Key = "INFORMATION"
      AND Keys.D# = Documents.D#
```

In SQL, we have to relate information spread over two relations, the Document relation and the Keys relation. In STAIRS, the information is structured so that the Keys are integrated with the rest of the data in a single file structure. The same query in AQL is:

```
   SELECT FROM Documents
      WHERE Key = "INFORMATION"
```

It is obvious that even at this simple level, the SQL query is much more difficult to formulate than the other two.

It is interesting to look at the same query where we want to retrieve parts of the documents such as the title, authors and associated keys. In STAIRS there is no difference. We simply specify the fields of interest when we print. However in SQL the query becomes much more complex, since the required information is spread over three relations.

```
SELECT Title Author Key FROM Documents, Keys, Authors
   WHERE Key = "INFORMATION"

   AND Keys.D# = Documents.D#
   AND Authors.D# = Documents.D#
```

In AQL, the query remains largely the same:

```
SELECT Title (Authors) (Keys) FROM Documents
   WHERE Key = "INFORMATION"
```

Parenthesisation is used to denote non-atomic objects, or lists. This example illustrates the main problem with the relational model, namely normalisation. Not only is the complexity of the query increased, but the object retrieved is itself not a "natural" one. Say there was one original document which satisfied the query - "THE ARRAY MODEL", with keys "INFORMATION" and "DATABASE" and authors "MACLEOD" and "BARNARD". Then the retrieved object which the user would see would be:

```
+-----------------+-------------+---------+
| THE ARRAY MODEL | INFORMATION | MACLEOD |
+-----------------+-------------+---------+
| THE ARRAY MODEL | DATABASE    | MACLEOD |
+-----------------+-------------+---------+
| THE ARRAY MODEL | INFORMATION | BARNARD |
+-----------------+-------------+---------+
| THE ARRAY MODEL | DATABASE    | BARNARD |
+-----------------+-------------+---------+
```

This is not a very aesthetically pleasing representation of what is wanted. It can be contrasted with the same information as represented in the array model:

```
+---------------+----------------------+-----------------+
|THE ARRAY MODEL|+-----------+--------+|+-------+-------+|
|               ||INFORMATION|DATABASE|||MACLEOD|BARNARD||
|               |+-----------+--------+|+-------+-------+|
+---------------+----------------------+-----------------+
```

This is a much more natural representation in that it corresponds almost exactly to the original document.

d) Find all documents containing the keys "INFORMATION" and "ARRAY"

```
SEARCH
   1: INFORMATION AND ARRAY
   2: 1.KEYS

SELECT D# FROM Keys
   WHERE Key = "INFORMATION"
   INTERSECT
   ( SELECT D# FROM Keys
     WHERE Key = "ARRAY"
   )

SELECT FROM Documents
   WHERE ("INFORMATION" IN Keys)
     AND ("ARRAY" IN Keys)
```

Boolean searches are one of the most common retrieval strategies. The SQL formulation is extremely awkward and very inappropriate for iterative searching.

e) Find the titles of all documents for which Barnard is the first author.

In STAIRS we apparently cannot do this, despite the fact that the information is implicitly available in the file organisation. Nor, in the relational model, can we answer this query with the three relations we have currently defined. However, if the Authors relation was modified to contain an additional attribute, position, then the query could be processed.

```
SELECT Title FROM Documents, Authors
   WHERE Author = "Barnard"
   AND Author.Position = 1
   AND Authors.D# = Documents.D#
```

A major benefit of using a data base approach to document retrieval, is that the information organisation is not so rigorously pre-determined. It is relatively straightforward to add a new attribute to a relation or to add a new relation to the database. In STAIRS we have little flexibility in defining the structure of the database. On the other hand, it would be difficult to argue that the above query is either an elegant or natural formulation of the query being posed. However the equivalent AQL query is completely straightforward.

```
SELECT FROM Documents
   WHERE FIRST Authors = "BARNARD"
```

f) Find all titles with more than three authors.

Another interesting class of queries involves quantification. This cannot be done in STAIRS, (although again, the necessary information is available in the file organisation). In SQL we can write:

```
SELECT D# FROM Authors
    GROUP BY D#
    HAVING COUNT (*) > 3
```

It is, once again, fairly obvious from this example that SQL does not lend itself to the straightforward expression of many types of simple queries. Essentially, what this query does is sort the relation by D# and count all the tuples with the same D#. In effect, non-atomic objects are being temporarily constructed. In contrast, the AQL equivalent can be expressed quite naturally.

```
SELECT FROM Documents
    WHERE TALLY Authors > 3
```

THE NEED FOR A NEW MODEL

Our view is that information is not naturally viewed and managed as a collection of sets. Rather information has structure and this structure is part of the information content. When we flatten information we lose part of the information itself. Our example on position illustrates this point. In most relational systems it is probably impossible to find the first author of an article. It can be said say that if this is important, a new relation can be created or position could be defined as an attribute. This seems unsatisfactory. It means a conscious effort has to be made to represent structure through attributes. That is, we have to predict potential queries which seems to violate the relational ideology. A typical document is not solely composed of atomic objects but rather may contain lists of objects, such as multiple authors, keywords, references and so on. These lists have properties such as length and order. The relational model tends to become unwieldy when we try to represent such an object in a purely tabular form.

Probably the major way in which bibliographic data differs from conventional data base retrieval is that retrieval is non-deterministic. That is, two different users might well expect to obtain different results for the same initial query. Much of the user's efforts in the bibliographic context is spent in interacting with the system in order to obtain the optimal group of documents related to the original query. In the conventional data base environment, the user typically spends more time in formulating the original query whereas in bibliographic data the query is continually reformulated or modified in response to the results obtained. Since continual viewing of the results is important in document retrieval, it is equally important that the structures implicit in the underlying data model reflect the structure of the information being retrieved. A basic problem with the relational model is that the internal representation can often be quite different from the external object. Furthermore, the intermediate object that must be constructed to rebuild the external object, is itself anomalous in that it is not optimally normalised. It is not natural to normalise data outside a computer information system and people don't usually do it.

While the incorporation of text handling features into the relational model in order to enhance its suitability as an information retrieval system is obviously useful, this approach should not be regarded as the final solution of the general problem of integrating information retrieval and data base systems.

In some ways the array model is a generalisation of the relational model. In fact, by restricting arrays to tabular structures, a relational representation is possible and there are many occasions when tables are natural structures. However, it is our strong contention that tables are not sufficiently powerful to provide a universal structure for all information. Structure is important and if structured objects can be handled in a relatively straightforward non-procedural manner, then there is no reason why we should not develop information systems based on such structures.

The array model appears to provide a natural basis for this type of approach. It is noteworthy that the structure of an AQL query is very simple. In fact it is similar to the material being retrieved. Where such a natural sort of mapping between the query structure and the structure of the information being retrieved, it seems apparent that the underlying DBMS is an appropriate model.

REFERENCES

1. Macleod, I. A. "SEQUEL as a Language for Document Retrieval", Journal of the American Society for Information Science. Vol. 30, pp. 243-249, 1979.

2. CRAWFORD, R. G. "The Relational Model in Information Retrieval", in Journal of the American Society for Information Science. Vol. 32, No. 1 (January 1981), pp. 51-64.

3. Macleod, I. A. "A Data Base Management System for Document Retrieval Applications, Information Systems, Vol. 6, pp. 131-137, 1981.

4. Schek, H.J. and Pistor, P. "Data Structures for an Integrated Data Base Management and Information Retrieval System", Proceedings of the Eighth International Conference on Very Large Data Bases. Mexico City, pp.197-207, 1982.

5. Chamberlin, D. and Boyce, R. "SEQUEL: A Structured English Query Language", Proceedings of the 1974 ACM-SIGMOD Workshop on Data Description, Access and Control. Ann Arbor, Michigan, (May 1974), pp. 249-264.

6. IBM Corporation. Storage and Information Retrieval, (STAIRS) Reference Manual.

7. Bibliographic Retrieval Services Inc. BRS Bulletin, Schenectady, New York.

8. More, T. "A Theory of Arrays with Applications to Databases" Tech. Report G320-2106, IBM Scientific Centre, Cambridge, Mass., 1975.
9. More, T. "Notes on the Diagrams, Logic and Operations of Array Theory", Tech. Report G320-2137, IBM Scientific Centre, Cambridge, Mass., 1981.

10. Jenkins, M.A. "The Q´Nial Reference Manual, Tech. Report 82-123, Queen´s University, Kingston, Ontario, 1982.

11. Macleod, I.A. "AQL - A Query Language for the Array Model", Technical Report, Department of Computing and Information Science, Queen´s University, Kingston, Ontario, 1983.

12. Macleod, I.A. A Model for Integrated Information Systems, Proceedings of the 9th Conference on Very Large Data Bases, pp.280-289, Florence, 1983.

COST FUNCTIONS FOR QUERY PROCESSING IN A DISTRIBUTED DATA BASE

P. Bodorik
Assistant Professor
School of Computer Science
Technical University of Nova Scotia
Halifax, Nova Scotia B3J 2X4

ABSTRACT

Various cost functions can be used in the formulation of a QPS in a DDB. The choice of a cost function depends on the application and the envirorment in which the DDB exists. An algorithm is described which creates a space of available strategies for processing of any given query. As the algorithm finds the cost of each strategy utilizing various measures/cost functions, it can be used as a tool in comparision and a choice of cost functions for any application of a DDB.

Les composantes du coût des recherches sur des bases de données réparties.

Différentes fonctions du coût peuvent être utilisées dans l'énoncé du QPS dans une DDB. Le choix d'une fonction de coût dépend de l'application et de l'environnement dans lequel la DDB existe. Un algorithme qui crée un espace de stratégies disponibles pour le traitement de n'importe quelle demande donnée est décrit. De même que l'algorithme trouve le coût de chaque stratégie en utilisant différentes fonctions mesures/coût, il peut aussi être utilisé comme outil dans la comparaison et le choix des fonctions de coût pour n'importe quelles utilisations de la DDB.

COST FUNCTIONS FOR QUERY PROCESSING IN A DDB

1. INTRODUCTION

Technological advances in computer networking stimulated a great interest in distributed processing for the purpose of sharing resources and data. Arguments of both a quantitative and qualitative nature were put forward in support of, and in opposition to, distribution; the problem became known as "Centralization vs. Decentralization" ((Reynolds, 1971), (Rockart, 1976)). A great interest has also been shown in the application of the concept of a Data Base Management System (DBMS) in a distributed environment supported by a network of geographically separate information processors. Users are provided with a unified data model, together with data description and manipulation languages, while the fact that data itself may is geographically distributed and supported by a network of cooperating information processors is hidden from users.

Advantages that can be gained by the use of a Distributed Data Base (DDB) in comparision to a centralized Data Base (DB) are ((Rothnie, 1977), (Toth, 1980)):

increased availablility;
improved response time;
modularity and upward scaling;
cost reduction.

A user specified query issued against a relational DDB is initially transformed either by a query processor or a pre-compiler, into relational algebra or relational calculus. In either case, the query is answered by execution of relational operators (selection, projection, join, union, etc.) on relations specified by the query. Since the ordering of operator execution is not fixed and relations are distributed over the network of information processors, before the query execution may begin the Query Processing Strategy (QPS) formulator must determine the sequence in which operators are executed, the location of their execution, and, of course, the movement of data/relations among processors. Formulation of the QPS comprises the query "strategy formulation" phase.

Once the QPS is determined, instructions are sent to local DBMSs for operator exection and transfer of data. The cooperation of local DBMSs by information exchange over the network, is necessary to process the query. The strategy's execution comprises the "query execution" phase.

COST FUNCTIONS FOR QUERY PROCESSING IN A DDB

LITERATURE REVIEW OF QPS IN A DDB

Optimization of a QPS in a central DB ((Smith, 1976), (Yao, 1978, 1979) to list a few) has led to a generally accepted set of rules (Ullman, 1982) which apply to a DDB as well. The rules are:

(i) One-variable operators (restrictions and projections) should be performed as early as possible as they reduce the size of relations which are operands of more costly two-variable operators.

(ii) Where applicable, one-variable operators should be performed (grouped/cascaded) together. Obviously, if there are several restrictions to be applied on one relation these should be applied simultaneously in one pass through the relation. For example, the query (R.a > 2) ^ (R.b < 4), where "^" is the "AND" boolean operator, specifies two restrictions on the same relation R. Clearly, both of these restrictions can be applied simultaneously on one pass through the relation with only an additional negligible cost of a comparision per tuple. Simultaneaous execution of such one-variable terms is called a cascaded one-variable term.

(iii) Similarly, if there are several two-variable terms specified for the same two relations, these should be applied simultaneously. Such terms are called cascaded two-variable terms or a cascaded join.

One of the earliest works on QPS optimization was done by Wong and Yossefi (1976). They decompose a query into a sequence of one-variable and two-variable queries and show that most of the time it is profitable to perform one-variable queries as soon as possible. A two-variable query is reduced into a set of one-variable queries by a tuple substitution. In other words, one tuple is taken from one relation at a time and applied to the other relation, thus creating a one-variable query. A two-variable query is in this way transformed by a tuple substitution into a sequence of one-variable queries. A modified version of this approach was adopted for a distributed version of the INGRES relational DB (Stonebraker, 1976) and was later evaluated on a test bed of queries (Youssefi, 1979).

In a network environment, the tuple substitution method to process two-variable terms may lead to unacceptable delays. To alleviate this problem, Epstein, Stonebraker, and Wong (1978) abandon tuple substitution and suggest that all local processing be performed first, followed by a feasible strategy which moves all results of one-variable processing to the destination node. This feasible strategy is improved upon by a "greedy" algorithm. The cost objective considered is either communication cost in terms of volume transmitted or CPU processing time which is approximated by the maximum CPU time delay for all CPUs involved in processing of the query.

The aforementioned algorithm was later evaluated by an experiment (Epstein, 1980) in which several key issues were investigated. The cost funtion used was the number of bytes transferred during query processing. Two conclusions were offered: (i) good size estimates are crucial, and (ii) a limited search through a space of query processing strategies performs poorly, suggesting that an exhaustive search for the optimal strategy is desirable.

Cellery and Meyer (1980) consider query processing in a multi-query environment. Their strategy's objective was minimizing the mean response time of queries, the only such attempt that could be found. Optimization is performed by the scheduling of data transfer and a local work-load on processors. A heuristic is proposed which optimizes queries' response time at the expense of redundant processing and communication.

A QPS for a DDB which uses a "virtual ring" method for concurrency control is discussed by Toan (1979). The proposed optimization method is based on a four-fold static optimization combined with threshold policies while using the control token circulation on the virtual ring of processors to solve conflicts of access between queries and updates.

In certain situations it may be advantageous to precede a join by a semi-join (Bernstein, 1979), a tactic borrowed from QPS optimization in a central DB (Smith, 1976). The method suggested by Henever and Yao (1979) is completely based on the use of semi-joins. They suggest that all joins should first be performed on appropriate projected "joining" attributes (ie., that semi-joins should be executed first), while whole relations are moved to the destination node concurrently. The results of semi-joins are also moved to the destination node where they are processed with the original relations to obtain the final result. They propose an algorithm which finds the optimal strategy for processing semi-joins for a restrictive set of "simple" queries. The algorithm is then extended to a heuristic which finds a sub-optimal strategy for processing of general semi-joins. The objective of the strategy is to minimize the total size of partial results, thus minimize the amount of transferred data.

Black and Luk (1982) propose a heuristic for generating a strategy for semi-joins. Its property is that for moderate size queries an exhaustive search may be performed.

Toth (1978, 1982) presents rules for the decomposition of queries. An optimal QPS is devised for a special class of sub-queries, Class A queries, which can be processed optimally. A Class A query is actually a superset of the simple query defined by Henever and Yao above. These sub-queries are processed first by an optimal strategy, thus reducing the size of intermediate results which are then processed by a sub-optimal strategy. The objective of the strategy is the minimization of the total size of data transferred under a maximum time constraint.

A combined static and dynamic decomposition of user queries for the MICROBE DDB is suggested by Toan (1981). Semi-joins are considered for query processing and a threshold mechanism with dynamic threshold policies is utilized for the correction of the strategy in case the cost of execution deviates from its estimate. Decentralized query management and a broadcast network is assumed. The cost measure used is the size of partial results.

The problems of RA and QPS were attacked simultaneously by Paik (1979). Large amounts of redundant data are avoided in order to minimize update synchronization of multicopy data and storage requirements. Assuming a minimum amount of redundant data optimally allocated over the network, a several-fold static optimization is performed on queries. Some dynamic optimization elements in query execution phase are introduced as well.

Query optimization in a star computer network is investigated by Kerscheberg, Ting and Yao (1982). The queries considered are assumed to be in a disjunctive normal form where each disjunction is a "sub-query" in a conjunctive normal form with each term having at most one two-variable operator. The QPS optimizes the query response time by minimizing secondary storage data access performed by information processors. The effect of communications speeds on the strategy's performance is investigated as well.

Bernstein et al. describe a QPS which was adopted for a System for Distributed Databases - SDD-1 (Bernstein, 1981). Queries are first translated into a relational calculus form called "envelope". There are two phases in the processing of a query. First, a query is reduced into several sub-queries by a decomposition process which was adopted from Youssefi (1979). Data for a particular sub-query is transferred to one chosen site where it is processed by use of semi-joins (reduction phase) and then assembled at one site for final processing. The cost function used is the size of operands (relations). Fragmented relations are considered only in the context of increased parallelism.

Relational algebra query processing optimization is considered by Chu and Hurlay (1979, 1982). A tree representing the original query's algebraic operations is constructed. Operations in this tree are then permutated to investigate alternate strategies. For that purpose, rules based on the permutability properties of relational algebra for identifying available/correct permutations are given. The objective of the optimization is the minimization of the volume of transferred data.

Fragmented relations are treated in a mathematical model by Pelagatti and Schreiber (1979). The model is targetted for application design stage, thus the speed at which a QPS for any query is arrived at is unimportant. The cost measure of the model is the data transfer cost. A framework for the quantitative analysis of performance based on various correlations between joining attributes, and between attributes and horizontal fragmentation, is provided. The model is capable of incorporation of redundant (multicopy) relations as well.

A special type of a query, a "set query", applied to relations, of which one may be horizontally distributed, is considered by Gavish and Segev (1982). They formulate the QP problem into a mathematical integer programming model and compare results of its solution by two heuristics. The cost function under consideration is the volume of transferred data.

OBJECTIVES

It is interesting to note that although different cost functions are claimed to be optimized by authors of various QPS formulators, in most cases the cost that is actually minimized is the sum of the size of partial results. This is assumed to minimize either the total volume of data processed by information processors or transferred over the network, depending on whether communication or CPU processing is considered to be the critical factor.

For example, in a distributed environment comprised of lower capacity CPUs connected by a high throughput local network, the total volume of processed data may be considered to be the critical parameter and hence it is a good candidate for adoption as a cost metric. It may be argued that by minimization of the size of partial results the total volume of processed data will be minimized as well, which in turn implies optimization of the response time.

Similarly, in an environment of high capacity processors which communicate by use of standard services provided by a public network, the communication delay may be considered to be the most critical factor. Minimization of the size of partial results will decrease the total volume of data transfer and hence decrease the query response time.

The aforementioned arguments ignore the potential for parallelism. A higher volume of processed or transferred data does not necessarily result in an increase in the response time, if the data is processed and/or transferred in parallel.

Matters are further complicated by the fact that trade-offs exist which depend on the choice of a cost function/measure. If the objective of the QPS formulation is the minimization of the query's total time delay, a higher volume of data transferred over the network and/or processed by CPUs, may be used to attain a higher parallelism in operations and thus a decrease in the query's time delay. In other words, a higher dollar cost is traded for a decrease in the time delay. Similarly, if the cost measure, used by the QPS formulator, is the dollar cost of communication, the optimal strategy may have a high cost in terms of the total time delay and the volume of data processed. In addition, various applications of a DDB imply different network and CPU processing unit costs.

It becomes apparent from the above discussion, that a particular cost measure may be suitable for one application but unsuitable for others. In an interactive environment, the time delay is considered to be the most critical factor and hence should be adopted as a measure of the query's cost, while in a batch environment a measure of throughput is more appropriate. Perhaps, the minimization of the size of query's partial results is a good compromise. To determine what cost function is to be used in the formulation of a QPS, a method/tool must exist that enables their comparision.

A relatively simple way to compare various cost functions for a particular application of a DDB, is to collect the set of expected queries and examine the cost of their execution using various cost measures while taking into consideration the frequency of their execution. This method, however, assumes that the optimal strategy and its cost can be determined for each query and cost function/measure under consideration. Optimal strategies can be determined from a space of strategies available for processing of a query, provided that each strategy has its cost calculated using various cost functions under consideration.

An algorithm, which creates the space of available strategies for any given query, is presented in Section 3. The space of strategies is represented by a graph organized in levels. Each graph node corresponds to a partial result obtained by a unique sub-strategy. The nodes of the final level represent unique strategy whose cost in various cost measures, is known. What cost functions are included is discussed in Section 4, which is followed by a summary.

2. ASSUMPTIONS

Assumptions on the query processing environment together with the terminology definition are now presented.

COST FUNCTIONS FOR QUERY PROCESSING IN A DDB

A. DDB Organization/Structure

Query processing is assumed to exist in the environment of a relational DDB based on, or similar to, ADD - Architecture for Distributed Databases - which utilizes a layered approach ((Mahmoud, 1979), (Toth, 1980, 1982)). Layers of interest are the Global view, Partitioned view and Distributed view layers.

In a relational DDB, relations may be fragmented vertically or horizontally (Mahmoud, 1979). Briefly, a horizontal fragmentation partitions a relation into subrelations according to a value of one partitioning key. All subrelations have mutually exclusive keys and identical attributes. Vertical fragmentation, on the other hand, is such that the key attribute(s) are replicated in all subrelations but the remaining non-key attributes are mutually exclusive. The cardinality of all sub-relations is therefore identical.

The user-issued query is transformed into the relational calculus conjunctive normal form defined on the global view of the DDB. The query becomes a conjunction of terms

T_1 ^ T_2 ^ T_3 ^ ... ,

in which each conjunction T_r is either a one-variable (restriction), or a two-variable (join) term. For example, the query (R_i.a > 5) ^ (R_j.b < R_k.c), where R_i, R_j and R_k are relations and a, b and c are attributes, contains two terms, the first is a one-variable restriction, while the second is a two-variable join. Relational arithmetic operators are drawn from a set of operators {=, >, <, >=, >=, =/=}.

The assumption that each term is at most two-variable is not restrictive since a term containing three or more relational variables can be transformed into a conjunction of two or more two-variable terms ((Ullman, 1980), (Epstein, 1979), (Mahmoud, 1979)). For example, a three-variable term (R_i.a > R_j.b = R_k.c) can be transformed into an equivalent conjunction of two two-variable terms (R_i.a > R_j.b) ^ (R_j.b = R_k.c).

A partitioned view processor defines the partitioning of global relations into vertical and horizontal fragments ((Chang, 1975), (Mahmoud, 1979)), or elementary relations, and it transforms the global view query to reflect this partitioning (Toth, 1980). A distributed view processor simply identifies the network locations of elementary relations.

B. Network and Static/Dynamic Strategy

A DDB is assumed to operate in the environment of a point-to-point network (e.g., ARPANET, DATAPAC). A QPS formulator is assumed to generate a static strategy. Once a strategy is formulated, it remains unchanged throughout the query exection phase.

C. Single/Multi Query Optimization

Strategies considered optimize strategies for a single query only; multi-query environment as proposed aby Cellery and Mayer (1981) is not considered.

D. Accepted Rules for QPS Formulation

Several generally accepted rules for QPS formulation, which were discussed in the literature review section, are adopted. In particular:

(i) Restrictions and projections are performed as soon as possible.
(ii) Cascaded one-variable operators are executed simultaneously.
(iii) Cascaded two-variable operators are executed simultaneously.

3. SPACE OF STRATEGIES

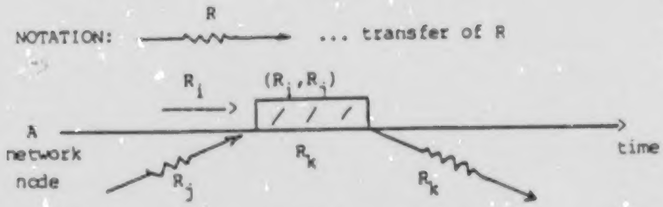
The problem of the creation of a space of strategies is addressed in this section. Before the algorithm for its creation is examined, examples of strategies which demonstrate that a QPS is defined not only by the sequence of joins to process two-variable terms but also by the locations where joins are executed, are given.

EXAMPLES OF QP

Assume that relations R_1, R_2, and R_3 are relations in network nodes A, B, and C respectively; a, b, ... are attributes; and θ_1, θ_2, ... are arithmetic relational operators. A simple query having only two two-variable terms $(R_1.a\ \theta_1\ R_2.b)$ ^ $(R_2.c\ \theta_2\ R_3.d)$ can be processed by various strategies. One of the strategies is:

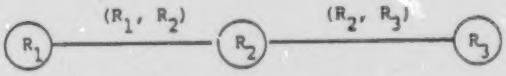
(i) join relations R_1 and R_2 in network node A to give relation R_{12},
(ii) join relations R_{12} and R_3 in network node C to give the query result.

The notation for timing diagrams is defined in Figure 1(a). The query and the timing diagram for the above strategy is shown in Figure 1(b) and (c) respectively. The timing diagram shows that relation R_1

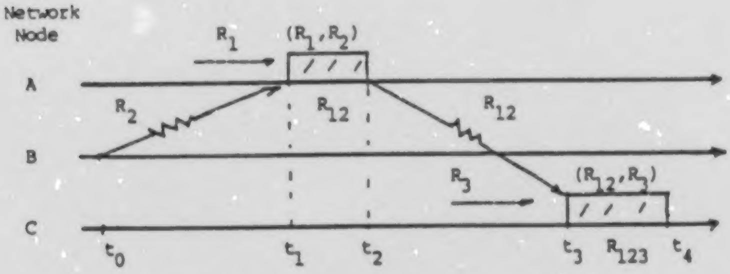

NOTATION:
R
... transfer of R
R_i
(R_i, R_j)
A
network
node
time
R_k
R_j
R_k

A join is executed to process a two-variable term ($R_i \theta R_j$). Relation R_i is located in network node A already, while relation R_j arrives just prior to the join execution of R_i with R_j. The join results in relation R_k which is then immediately transferred to some other node.

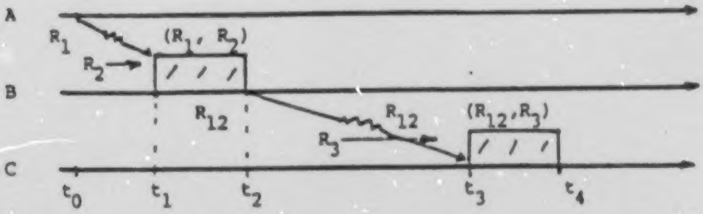
1(a) Notation for strategy timing diagrams


R_1
(R_1, R_2)
R_2
(R_2, R_3)
R_3

1(b) Query ($R_1.a \theta_1 R_2.b$) $\wedge$ ($R_2.c \theta_2 R_3$)


Network
Node
A
B
C
R_1
(R_1, R_2)
R_2
R_{12}
R_{12}
R_3
(R_{12}, R_3)
t_0
t_1
t_2
t_3
R_{123}
t_4

1(c) Strategy timing diagram


A
B
C
R_1
R_2
(R_1, R_2)
R_{12}
R_{12}
R_3
(R_{12}, R_3)
t_0
t_1
t_2
t_3
t_4

1(d) Strategy timing diagram

Figure 1 A query and two strategies for its processing

is transferred to node B and immediately upon its arrival at time t_1, the join of relation R_1 and R_2 commences to process the two-variable term ($R_1.a \theta_1 R_2.b$). The join is completed at time t_2 giving relation R_{12}, which is then transferred to node C. When this transfer is completed at time t_3, the join of R_{12} with R_3 to process ($R_{12}.c \theta_2 R_3.d$) commences. The query result, relation R_{123}, is available in node C at time t_4.

The second strategy, whose timing diagram is shown in Figure 1(d), may be:

(i) join relations R_1 and R_2 in network node B to give relation R_{12},
(ii) join relations R_{12} and R_3 in network node C to give the query result.

Both strategies are identical in the sequence of join execution. They both specify that first R_1 is to be joined with R_2 giving R_{12}, followed by the join of R_{12} with R_3 in node C to give the query result, while their only difference is in the location of the first join is execution. One specifies node A, while the other node B.

If it is assumed that the processors in nodes A and B have an identical load and are of the same capacities, ie., they have the same processing power and incure the same delay due to access to secondary storage devices, then the delay caused by the processing of a join of R_1 with R_2 in node A will be identical to that of the same join but executed in node B. However, even under this assumption, the two strategies produce different time delays. If R_1 and R_2 are of a different size, which is likely, then the transfer cost of R_1 from node A to B, as specified by the first strategy, will differ from the cost of transfer of R_2 from node B to A which is specified by the second strategy.

In addition to assumptions of Section 2, the following assumptions are made by the algorithm which builds a space of strategies.

a) The first assumption is made on a priori cost and size estimation of partial resutls. Given two relations R_i and R_j, an algorithm is able to estimate the size and cost of the partial result of joining R_i with R_j to process a two-variable term ($R_i.a$ 0 $R_j.b$). Relations R_i and R_j can be either elementary relations referenced by the query, or estimates of partial results of processing of some two-variable terms. Included in the estimation is the removal of joining attributes (attributes 'a' and 'b' for the above relations R_i and R_j) by a projection. Projection is included only in the case, of course, when these attributes are not 'target' attributes and are no longer necessary for execution of some other joins necessary to process the query.

b) The second assumption refers to the destination of the query result. When a query is issued against the DDB, the desired destination, to which the query result should be delivered by the DDB, is specified as well. The destination node is typically, but not necessarily, that node in which the query is issued. The strategy for processing of a given query must be such that the query result is delivered to the specified destination node. It is assumed that when a query processing strategy under examination is such that the query result is not obtained in the destination node, then the strategy is augmented by the final transfer of the result to the destination node. The total cost of a strategy is increased by the cost of this final data transfer.

c) The third assumption specifies that a join of two relations R_i and R_j can be executed only in a network node where either R_i or R_j is located.

d) Finally, it is assumed that relations referenced by a query are not partitioned either horizontally or vertically. A query may therefore be considered to be expressed on the global view of the data base (Toth 1980, 1982).

CREATION OF A SPACE OF STRATEGIES

The development of a space of strategies for a given query can be demonstrated by the following example.

Consider a simple query $(R_1.a > R_2.b) \wedge (R_2.c < R_3.d)$ referencing relations R_1, R_2, and R_3 located in network nodes A, B, and C respectively. The space of strategies must be such that it reflects a choice of the sequence of joins and their locations of execution. Options for the selection of the first join, together with the location of its execution, are:

(i) ${}_{1}R^{A}_{01}$ and ${}_{2}R^{B}_{02}$ are joined in A (ie., ${}_{2}R^{B}_{02}$ is moved to node A) to give relation ${}^{01,02}_{12}R^{A}_{11}$

(ii) ${}_{1}R^{A}_{01}$ and ${}_{2}R^{B}_{02}$ are joined in B to give ${}^{01,02}_{12}R^{B}_{12}$

(iii) ${}_{2}R^{B}_{02}$ and ${}_{3}R^{C}_{03}$ are joined in B to give ${}^{02,03}_{23}R^{B}_{13}$

(iv) ${}_{2}R^{B}_{02}$ and ${}_{3}R^{C}_{03}$ are joined in C to give ${}^{02,03}_{23}R^{C}_{14}$

The labelling scheme for relations, subscripts and superscripts, will be presented shortly.

The above options can be represented by a directed graph in which nodes represent either elementary relations, ie., relations referenced by the query, or relations which are results of some join executions to process two-variable terms. Edges outgoing from a graph node identify relations from which it was derived by an execution of a join.

To avoid the confusion between network nodes and nodes of a graph, the graph nodes will be referred to as graph elements, or as elements. Since the graph nodes, or elements, represent relations they will also be sometimes referred to simply as relations so that the repetition of the phrase "element representing a relation" is avoided.

Figure 2(a) shows the above options represented by a directed graph. As the elementary relations are not results of any join execution, the elements representing them do not have any outgoing edges. The graph elements representing results of the first join, on the other hand, have two outgoing edges each. Edges identify relations from which the graph element was derived by a join. The graph elements are labelled using the notation ${}^{lsp}_{lsb}R^{rsp}_{rsb}$, in which the left and right subscripts and superscripts have the following meaning:

(i) RIGHT SUPERSCRIPT identifies a network node in which the relation is located by a use of capitals A, B, C, ...

(ii) RIGHT SUBSCRIPT is the identifier, or ID. It consists of two parts. The first part is the number of joins executed to derive the relation, while the second one is a number which is unique for all elements derived by the particular number of joins. For instance, if two relations are derived by exactly two joins, the first part (which is actually the first digit of their right subscript) will be 2, while their second parts (consisting of the remaining digits) will be different. It is tacidly assumed that any query can be processed by at most 9 joins, but this notation will suffice for exposition purposes. If an element represents an elementary relation referenced by the query, then the first digit of its ID is 0, as the relation is not a result of any join. Note that a query definition may specify relations R_1, R_2, ..., up to R_9 only, but again, this is deemed to be sufficient. There is a direct correspondence between the query definition and the directed graph representing the space of strategies. An elementary relation, ie., relation originally referenced by the query, is denoted by R_i, i=1, 2, ..., 9 in the query specification, while in a directed graph it is represented by an element with a label ${}^{lsp}_{lsb}R^{rsp}_{0i}$, ie., its right subscript is the same, only it is preceeded by a "0".

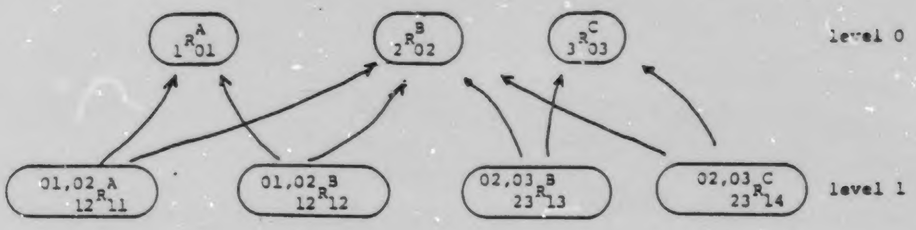

1 R A 01
2 R B 02
3 R C 03
level 0
01,02 12 R A 11
01,02 12 R B 12
02,03 23 R B 13
02,03 23 R C 14
level 1

Figure 2(a) Graph Nodes Representing Elementary Relations (Level 0) and Results of One Join (Level 1)

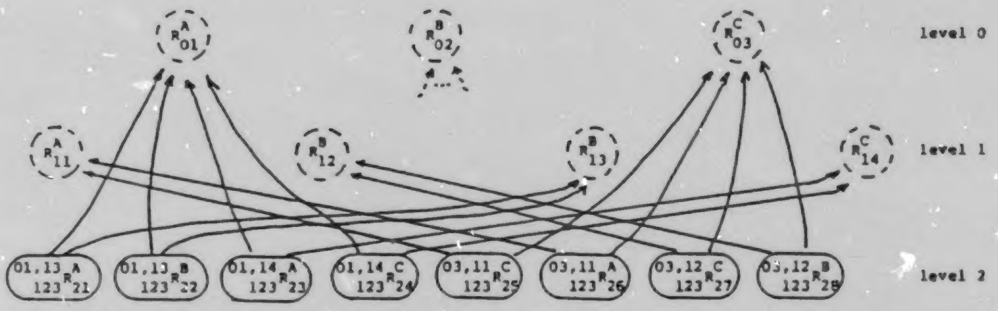

R A 01
R B 02
R C 03
level 0
R A 11
R B 12
R B 13
R C 14
level 1
01,13 123 R A 21
01,13 123 R B 22
01,14 123 R A 23
01,14 123 R C 24
03,11 123 R C 25
03,11 123 R A 26
03,12 123 R C 27
03,12 123 R B 28
level 2

Figure 2(b) Elements Representing Relations which are Results of Two Joins (Level 2)

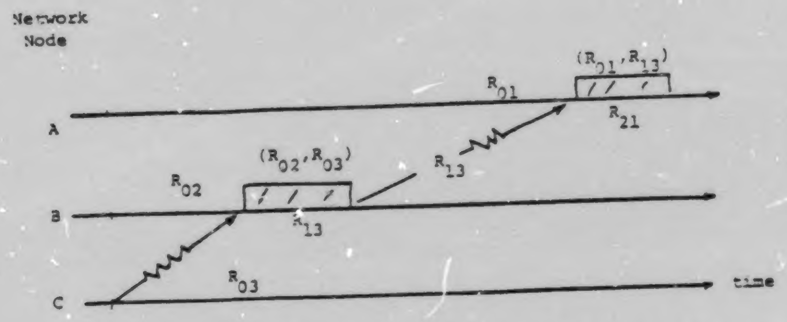

Network Node
A
B
C
R_{01}
(R_{01}, R_{13})
R_{21}
R_{13}
(R_{02}, R_{03})
R_{02}
R_{13}
R_{03}
time

Figure 2(c) Strategy Timing Diagram for ${}^{01,13}_{123}R^{A}_{21}$

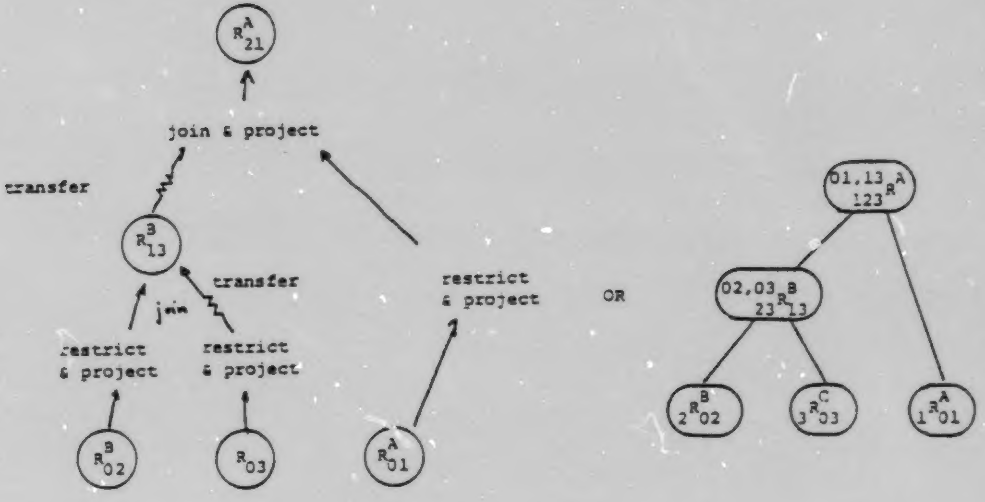

R^{A}_{21}
join & project
transfer
R^{B}_{13}
join
transfer
restrict & project
restrict & project
restrict & project
R^{B}_{02}
R_{03}
R^{A}_{01}
OR
${}^{01,13}_{123}R^{A}$
${}^{02,03}_{23}R^{B}_{13}$
${}_{2}R^{B}_{02}$
${}_{3}R^{C}_{03}$
${}_{1}R^{A}_{01}$

Figure 2(d) Tree of Elements Representing a Strategy for ${}^{01,13}_{123}R^{A}_{21}$

(iii) LEFT SUPERSCRIPT consists of a pair of ID's of relations processed by a join to create the element. For each of the two IDs in the left superscript there is an outgoing edge to the element with that ID, ie., to an element with a right subscript matching the ID.

(iv) LEFT SUBSCRIPT identifies relations processed by joins to derive the relation represented by the element. If the relation is a result of a join of relation R_1 with R_2, followed by a join of the result with R_3, then the left subscript will be 123.

For instance, if the element is ${}^{121,03}_{123}R^{A}_{21}$, then the element represents a relation located in network node A. Its ID is 21 signifying that the relation is a result of 2 joins. Because the left subscript is 123, the relation is a result of joins which process elementary relations R_1, R_2, and R_3. The relation is a result of a join of two relations represented by elements with ID's 121 and 03.

Continuing with the example, consider relations which are results of exactly two joins:

(i) ${}_{1}R^{A}_{01}$ is joined with ${}^{02,03}_{23}R^{B}_{13}$ in node A to give ${}^{01,13}_{123}R^{A}_{21}$

(ii) ${}_{1}R^{A}_{01}$ is joined with ${}^{02,03}_{23}R^{B}_{13}$ in node B to give ${}^{01,13}_{123}R^{B}_{22}$

(iii) ${}_{1}R^{A}_{01}$ is joined with ${}^{02,03}_{23}R^{C}_{14}$ in node A to give ${}^{01,14}_{123}R^{A}_{23}$

(iv) ${}_{1}R^{A}_{01}$ is joined with ${}^{02,03}_{23}R^{C}_{14}$ in node C to give ${}^{01,14}_{123}R^{C}_{24}$

(v) ${}_{3}R^{C}_{03}$ is joined with ${}^{01,02}_{12}R^{A}_{11}$ in node C to give ${}^{03,11}_{123}R^{C}_{25}$

(vi) ${}_{3}R^{C}_{03}$ is joined with ${}^{01,02}_{12}R^{A}_{11}$ in node A to give ${}^{03,11}_{123}R^{A}_{26}$

(vii) ${}_{3}R^{C}_{03}$ is joined with ${}^{01,02}_{12}R^{A}_{12}$ in node C to give ${}^{03,12}_{123}R^{C}_{27}$

(viii) ${}_{3}R^{C}_{03}$ is joined with ${}^{01,02}_{12}R^{A}_{12}$ in node B to give ${}^{03,12}_{123}R^{B}_{28}$

The graph representation for the above options is derived from a graph shown in Figure 2(a). For each relation which is a result of two joins, a new element with two outgoing edges is added to the graph. Edges of any new element terminate at relations processed by a join to derive the element. The directed graph is shown in Figure 2(b).

DIRECTED GRAPH

Assume the usual definition of a "sink" node - it is a node with no outgoing edges. In a directed graph representing a space of strategies, a sink node always represents one of the elementary relations. Elements which represent the query result obtained by different strategies have no incoming edges, call such elements "final". Given a final element, its strategy can be obtained simply by following its outgoing edges. For the example of Figure 2(b), the element ${}^{01,13}_{123}R^{A}_{21}$ identifies a strategy shown in Figure 2(c), which is determined by following all the paths from the final node to the sinks.

Obviously, the graph is connencted, there is at least one incoming or outgoing edge for every element. A graph element/node is either (i) a sink representing an elementary relation referenced by the query, in which case there must be at least one incoming edge as the relation must be processed by a join, or (ii) an element representing a relation which is a result of some processing and therefore it must have outgoing edges.

Definition: Level p of the directed graph representing the space of available strategies is a set of nodes which represent relations processed by exactly p joins.

Certain observations, actually theorems presented without their obvious proofs, that can be made about the graph properties follow.

OBSERVATIONS:

Observation 1: Levels are mutually exclusive sets of graph nodes.

Observation 2: Every node of the graph is in exactly one level.

Observation 3: The graph has levels 0, 1, 2, ..., n_j, where n_j is the number of joins necessary to process a query, ie., the graph has $n_j + 1$ levels.

Observation 4: A node of level $p > 0$ has exactly two outgoing edges to nodes of level p_x and p_y, such that $p_x, p_y < p$ and $p=p_x+p_y+1$.

Observation 5: Every node, which is not a final node, must have at least one incoming edge.

Observation 6: Every final node, ie., a node representing the query result obtained by a particular strategy, is a root of a binary tree which defines the strategy to derive it. The tree representing the strategy of Figure 2(c) is shown in Figure 2(d). Note that this is the usual representation of a QPS appearing in the scientific literature on QPSs.

NUMBER OF LEVELS / NUMBER OF NECESSARY JOINS

Each element of the directed graph at level p represents a relation which is a result of processing of p joins. The number of levels in a graph, therefore, depends on the number of joins which are necessary to process a given query, while it, in turn, depends on the number of two-variable terms in the query definition and the number of query's cascaded joins. However, a much simpler relationship exists for a global view query (Bodorik, 1984) which implies that if the number of relations refernced by a query is n_r then the number of joins, some of which may be cascaded, to process the query is $n_j = n_r-1$.

The number of joins which is necessary to process a query is not only of general interest, but it is also used by the following algorithm which creates the space of available strategies.

ALGORITHM

Observation 6 infers that a directed graph representing the space of available strategies can be viewed as a collection of binary trees, each representing one QPS, such that the trees may "share" some of their nodes. The problem is how to construct the graph of strategies for a given query in an efficient way and without any redundancies. In other words, the graph should be such that each of its elements represents a partial result derived by a unique strategy. If such a directed graph is constructed for a query and the cost of relations represented by the graph elements is known, then it is a simple task to examine all final nodes to find the one with the minimum cost. It identifies the query's minimum cost strategy, while the strategy itself is defined by a tree of which the minimum cost node is a root.

The example of Figure 2 demonstrates how the algorithm constructs the directed graph level by level. Initially, level 0 is built from the elementary relations. The cost of a relation of level 0 includes the cost of one-variable processing, ie., the cost of restrictions and a projection to remove attributes which are not necessary for further processing. Representation of possible ways to create a partial result of exactly two joins is created from the graph of Figure 2(a) by an addition of another level as shown in Figure 2(b). An element of a new level p is created from two elements of the previous levels p_x and p_y such that $p = p_x + p_y + 1$. In addition, elements of levels p_x and p_y must be such that they represent relations that can be joined. For instance, in Figure 2(a), R_{12} and R_{01} can not be joined, neither can R_{01} and R_{03}.

Paraphrasing, a relation R, which is a result of p joins and is therefore an element of level p, can be created by a join of two relations R_i and R_j of levels p_x and p_y respectively, where R_i is a result of p_x joins, while R_j of p_y joins. If R_i and R_j are joined to give R then R will be a result of $p_x + p_y + 1$ joins. As a result, relationship $p = p_x + p_y + 1$ must be satisfied.

To summarize, the graph construction is started by the creation of level 0, whose nodes represent original relations referenced by the query. Then level 1, 2, ..., n_j are built in that order, where n_j is the number of joins necessary to process a given query. Nodes of level p represent relations which are results of exactly p join executions. An element of level p is constructed from elements of levels p_x and p_y such that $p = 1 + p_x + p_y$. The number of graph levels is equal to the number of joins necessary to process a query. Any time a new element representing relation R is constructed from elements representing relations R_i and R_j, its cost and size is estimated. This conforms to the assumption (a) of this section. In addition, pointers to nodes representing R_i and R_j are added, they represent graph edges to relations from which the new node is created.

An element of the last level represents the query result obtained by a particular strategy. If this result is not located in the destination node, the element's cost is increased by the cost of transfer to the destination node and the strategy is augmented by this transfer as well. This comforms to the assumption (b).

Since each element of the last level represent the query result obtained by a different strategy, to find the optimal strategy, the last level is searched for the node with the minimum cost. The optimal strategy is found by following the pointers from the minimum cost element to the sinks representing query's original relations.

An algorithm, which constructs a graph of available strategies for a given query, is shown in Figure 3. It creates a graph representing all available strategies and, in addition, the graph is such that it does not contain any redundant elements. The redundancy is avoided by taking into consideration all of the available information any time a new graph element, which represents a sub-strategy, is created.

PARTITIONED RELATIONS

The algorithm of Figure 3 assumes that the query is expressed on the glogal view of the DB. This assumption implies that a query does not reference relations which are partitioned either horizontally or vertically. Since realistic applications of a DDB necessitate the use of partitioned relations, this assumption must be removed in order to allow a query to be expressed on the distributed view of the DB, thus

INPUT: Query Q defined by $\{(R, L, INFO)_i\}$ and $\{2\text{-}V_u\}$, where

- (i) $\{2\text{-}V_u\}$, u = 1, 2, ... is a set of two-variable terms for query Q, and
- (ii) $\{(R, L, INFO)_i\}$, i = 1, 2, ... is a set of quadruples conveying information about relations referred to by the two-variable terms, wherc
  - R_i is a relation referred to by a two-variable term in $\{(2\text{-}V)_u\}$
  -
  - L_i is the network location of R_i
  - $INFO_i$ is information about relation R_i which is necessary for size and cost estimation of join processing.

OUTPUT: The graph of QP strategies and their costs. QPS are

Method: The graph of strategies is created by levels. Each graph element represents either the original relation referenced by the query (elementary relation) or a relation which is a partial result of processing of one or more joins. Level 0 represents elementary relations with processed restrictions and projections that remove unnecessary attributes (ie., attributes that are neither target or joining attributes), level 1 represents relations which are partial results of processing exactly one join, and level p represents relations which are partial results of exactly p joins. Each graph node contains:

- two pointers to its predecessors from which it is created
- pointer to the next element of the same level
- quadruple conveying information about the relation the node represents, ie., if the relation is R_i then the quadruple is $(C, L, \{R\}, INFO)_i$, where
  - C_i is the cost of relation R_i
  - L_i is the network location of R_i
  - $INFO_i$ is information about relation R_i which is necessary for size and cost estimation of join processing.
  - $\{R\}_i$ is a list of elementary relations processed by joins to derive R_i. For example, if R_i is a result of a join of R_1 with R_2, followed by the join of the result with R_3, then $\{R\}_i = \{R_1, R_2, R_3\}$.

Note the correspondence of the information stored in each graph node and the labelling scheme for graph elements in pictorial representation representation of the directed graph. For relation R_i at level p and its quadruple $(C, L, \{R\}, INFO)_i$, the right subscript of its label is "i" preceded by "p", ie., "pi". The right superscript is L_i, the location of the relation. The left superscript corresponds to the two graph edges - the graph node pointers. The left subscript corresponds to $\{R\}_i$. If $L_i=\lambda$, $\{R\}_i = \{R_1, R_2\}$, i=7, and the relation is at level 1, then in the pictorial representation of the directed graph, the graph node will be labelled ${}^{01,02}_{12}R^{\lambda}_{17}$.

STEPS

1. Create level zero of the graph of strategies directly from relations referenced by the query and set the cost of each element to the cost of its restrictions and projections.

2. Do steps (2) to (4) for p = 1, 2, ..., n_j, to create levels 1, 2, ..., n_j, where n_j is the number of necessary joins to process the query.

3. Create level p from elements of previous levels by execution of step (4) for $p_x = p-1, p-2, \ldots |p/2|$.

4. For each tree element R_i of level p_x, search level p_y, such that $p = p_x + p_y + 1$, for those elements which represent a relation which can be joined with R_i. For each such found element R_j, create a new element which represents relation R_k arrived at by a join of R_i with R_j in the network node where R_i is located. Set the quadruple $(C, L, \{R\}, INFO)_k$ of the new element representing relation R_k to:
   - C_k is set to the cost of R_k
   - L_k is set to L_i
   - $\{R\}_k$ is set to the union of $\{R\}_i$ and $\{R\}_j$
   - $INFO_k$ is derived from $INFO_i$ and $INFO_j$

   The pointers of the new nodes are set to identify the elements representing R_i and R_j. Finally, the new node is inserted to level p. If p_x =/= p_y and L_i =/= L_j (ie., R_i and R_j are not located in the same network node) then create an additional node which represents the join of R_i with R_j in network node L_j where R_j is located and insert it to level p.

Figure 3 Algorithm to develop a graph of available strategies

BEST COPY AVAILABLE

allow partitioned relations. The algorithm was extended to process relations which are partitioned horizontally and/or vertically and it can be found in (Bodorik, 1984) - it is not discussed here any further.

4. COST FUNCTIONS

The algorithm of Figure 3 does not specify what cost function is utilized to determine the cost of partial results and strategies. Its only assumption is that given two relations and information about them, ie., their cost, cardinalities, locations and other information necessary for estimation of the size of partial result, there is a cost function which evaluates the cost of the join operation and the cost of the data transfer. The cost of CPU processing and data transfer can be obtained with reasonable accuracy provided that the size of relations is known; it will depend on the unit costs for CPU processing and data transfer.

Futhermore, each partial result and strategy can have its cost determined in more than one cost measure simultaneously. In fact, the algorithm was implemented (Bodorik, 1984) to calculate the cost of each partial result using the following measures:

total time delay;
time delay due to CPU processing only;
time delay due to communication only;
total dollar cost (depends on the volume of data processed and transferred);
dollar cost due to the CPU processing only (depends on the volume of data processed);
dollar cost due to the data transfer only (depends on the volume of data transferred);
sum of the sizes of partial results.

Since the algorithm develops the space of strategies in which the cost of each strategy is given in various cost measures, optimal strategy for each of the measures can be found by a search through the final level of the graph. Once a minimum cost strategy is found for one cost measure, for example for the total time delay, the strategy's cost using other measures, for instance, the total dollar cost, is found as well. As a result, the algorithm can be used as a tool for investigation on the choice of a cost function for processing of queries.

5. CONCLUSION

Various cost functions can be used in the formulation of a QPS in a DDB. The choice of a cost function depends on the application and the environment in which the DDB exists. The query's cost depends not only on the delays and dollar unit costs due to CPU processing and data transfer, but also on the strategy formulation which minimizes the query's cost using a particular cost function.

An algorithm was described which creates a space of available strategies for processing of any given query. As the algorithm finds the cost of each strategy utilizing various measures/cost functions, it can be used as a tool in comparision and a choice of cost functions for any application of a DDB.

The algorithm was implemented and will be tested on several applications of a DDB in the near future.

REFERENCES

Bernstein, P., A., "Approaches to Concurrency Control in Distributed Data Base Systems", AFIPS, 1979, 813-820.

Bernstein, P. A., et. al., "Query Processing in a System for Distributed Databases (SDD-1)", ACM TDS, Vol. 6, No. 4, Dec. 1981, 602-625.

Black, P. and Luk, W. S., "A New Heuristic for Generating Semi-Join Programs for Distributed Query Processing", Proc. of COMPSAC 82, IEEE Computing Society's 6th International Conf., 1982, 581-588

Bodorik, P., "Query Processing Strategies in a Distributed Data Base", Ph. D. dissertation (in preperation), Carleton University, Ottawa, 1984 (expected).

Cellary, W., and Meyer, D., "A Multi-Query Approach to Distributed Processing in Relational DBMS", Distributed Data Bases, North-Holland Publishing Company, 1980, 99-119.

Chang, S-K. and Cheng, W-H., "A Methodology for Structured Database Decomposition", IEEE TOSE, March 1980, Vol. SE-6, NO. 2, 205-218.

Chu, W. W. and Hurley, P., "A Model for Optimal Query Processing for Distributed Data Bases", CG1393-8/79, 1979 IEEE, 116-122.

Epstein, R., et. al., "Distributed Query Processing in a Relational Data Base System", ACM-SIGMOD, Proc. International Conf. on Management of Data, Austin, Texas, 1978, 169-180.

Epstein, R. and Stonebraker, M., "Analysis of Distributed Data Base Processing Strategies", 6th International Conf. on Very Large Data Bases, Montreal, Quebec, 1980, 92-101.

Gavish, B. and Seger, A., "Query Optimization in Distributed Computer Systems", Management of Distributed Data Processing, North Holland Publ. Comp., 1982, 233-252.

Henever, A. R. and Yao, S. B., "Query Processing in Distributed Data Base Systems", IEEE TOSE, vol. SE-5., No. 3, May 1979, 177-187.

Kerschberg, L., Ting, P. D. and Yao, B., "Query Optimization in Star Computer Networks"., ACM TDS, Vol. 7, No. 4, Dec. 1982, 678-711.

Mahmoud, S. A., Riordon, J. S. and Toth, K. C., "Distributed Database Partitioning and Query Processing", IFIP-TC-2, Venice, Italy, 1979, 32-51.

Paik, I-S and Delobel, C., "A Strategy for Optimization of Distributed Query Processing", Proc. 1st Int. Conf. on Distributed Computing Systems, Huntsville, Alabama, Oct. 1979, 686-698.

Pelagatti, G. and Schreiber, F. A., "Evaluation of Transmission Requirements in Distributed Database Access", ACM-SIGMOD 1979, Int. Conf. on Mamagement of Data, May 1979, Boston, Mass., 102-108.

Reynolds, C. H., "Issues in Centralization", Datamation, March 1977.

Rockart, J. F. and Leventer, J. S., "Centralization vs. Decentralization", Working Paper WP-845-76, April 1976, MIT Sloan School of Management, Cambridge, Mass., USA.

Smith, J. M. and Chang, P. Y., "Optimizing the Performance of a Relational Algebra Database Interface", CACM, Vol. 18, No. 10, Oct. 1975, 568-579.

Stonebraker, M. and Neuhold, E., "A Distributed Database Version of Ingres", Memorandum No. ERL-M612, Sept. 1976, Eng. Research Lab., University of California, Berkeley, California.

Toan, N. G., "A Unified Method for Query Decomposition and Shared Information Updating in Distributed Systems", First Int. Conf. on Distributed Computer Systems, Huntsville, Alabama, Oct. 1979, 679-685.

Toan, N. G., "Distributed Query Management for a Local Network", Proc. 2nd Int. Conf. on Distributed Computing Systems, Paris, France, April 1981, 188-196. Toth, K. C., "Distributed Database Architecture and Query Processing", Ph. D. Dissertation, Carleton University, Ottawa, Ontario, 1980.

Toth, K.C., et. al., "Query Processing Strategies in a Distributed Database Architecture", Distributed Data Sharing Systems, North Holland Publ. Comp., 1982, 117-134.

Ullman, J. D., "Principles of Database Systems", Computer Science Press, 1982, 2nd edition.

Wong, E. and Youssefi, K., "Decomposition - A Strategy for Query Processing", ACM TDS, Vol. 1., No. 3., Sept. 1976, 223-241.

Youssefi, K. and Wong E., "Query Processing in a Relational Database Management System", IEEE, CH1406-8/79, 407-417.

Yao, S. B., "Optimization of Query Evalution Algorithms", ACM TDS, Vol. 4, No. 2, June 1979, 133-155.

REWARDS AND PITFALLS OF TURNING TOWARD A MICRO (AND AWAY FROM INTEGRATED SYSTEMS) FOR ACQUISITIONS: THE IDEAL VS. THE AFFORDABLE

Mary E. Campbell
Librarian for Co-ordination of Automated Processes
R.W.B. Jackson Library
Ontario Institute for Studies in Education
252 Bloor St. W., Toronto Ont. M5S 1V6

In Collaboration with
L.F. "Buddy" Doll
Senior Consultant
Micro Software Sales
120 Norfinch Dr., Unit 37
Downsview, Ont. M3N 1X3

ABSTRACT

While the professional literature has come more and more to urging the acceptance of integrated/compatible library systems rather than stand-alone ordering, cataloguing and circulation etc. components, the cost of this interaction may not be feasible or justifiable in today's climate of fiscal constraint. The decision-making process of a medium-sized academic library which led to the choice of a microcomputer-based non-integrated ordering and accounting system is examined in this paper; the resulting system, its strengths and weaknesses, and the trade-offs necessary in it, is outlined and explored.

Avantages et inconvénients du microordinateur (en dehors d'un système intégré) pour les acquisitions.

Alors que l'on prône de plus en plus, dans la littérature professionnelle, l'utilisation de systèmes automatisés intégrés pour les bibliothèques plutôt que des modules indépendants d'acquisition, de catalogage et de circulation, le coût de cette intégration peut s'avérer prohibitif et injustifiable dans le contexte actuel des restrictions budgétaires de toutes sortes. Les auteurs présentent et discutent la décision d'une petite bibliothèque universitaire de choisir un microordinateur pour gérer un système non intégré d'acquisition et de comptabilité. Enfin, ils analysent le système, ses forces, ses faiblesses et les compromis qu'il exige.

MICRO BASED ACQUISITIONS

> "The question then is: What price integration?"
> (Boss, 1980, p. 162)

This question, asked four years ago in the keynote address at the LITA Institute on automated acquisitions, is often the crux of the problem for many libraries which are still exploring automation options for their major functions. Automation of various library procedures has been underway for two decades now. Activities ranging from accounting, to acquisitions tasks, to bibliographic data, booking systems for media centres, budget control and generation, cataloguing and catalogue information, circulation control, collection management, correspondence, indexing, information retrieval, interlibrary loan, inventory control, on-line searching, personnel records, purchasing, serials and binding control, statistical reports, to word processing (to name but a few), have all been candidates for conversion to machine handling during this time (see Beaumont and Kreuger, 1983, pp. 119-122).

Libraries which started in the 60s used large computers, often under the control of other departments of their parent institutions, and largely developed their systems using either the computer staff of those facilities (who often lacked an understanding of library goals, tasks and routines) or attempted to use their own staff who similarly often lacked the computer expertise. Many a library found itself as a second-class user with little priority on the time and efforts of those in control of this equipment. But the search for the perfect integrated system was already on. In the 70s the technology rapidly progressed, and the Utilities grew up, mainly out of the efforts of individual libraries to solve their own problems, who then branched out to include others outside their original scope. Librarians joined the utilities

> "...because they saw them as the only technically and economically feasible way to cope with the rapidly increasing rates of library costs and even the smaller libraries could benefit...Joining a network was an inexpensive and risk-free way of using technology"

to cope with declining resources and staff. (De Gennaro, 1983, p. 631).

But the realities of the eighties' economics are even harsher - rapid inflation has hit the costs of participating in these networks, and as their complexities grow, and more and more functions can be done through them, costs have become very considerable. As networks falter in their ability to deliver services at favourable prices, members are being forced to seek other alternatives. The utilities are in danger of being seen as inflexible and unresponsive to the growing and changing needs of the multitype library user. And the availability of alternatives makes the eighties a potentially exciting time to explore options. A library or group of libraries may now choose, theoretically, from a shopping list of such solutions as direct vendor/library computer interaction; use of the utilities for searching and cataloguing as well as vendor-direct interaction; locally based turnkey systems for such major tasks as circulation control, on-line catalogues, acquisitions and serials control; and microcomputer-based modules of any of these, either stand-alone or interfaced with other systems which might be local or shared. Depending on a library's size, any or all of these solutions may be selected and ideally connected to each other directly. Decentralization is again the movement. (Again, see De Gennaro, 1983)

So, starting in the sixties, with primitive, stand-alone, local, mainly batch systems, the shift was to the large multifunction bibliographic utilities such as UTLAS, OCLC, RLIN and WLN in the seventies; in the eighties has come the trend to "decentralization with integration", whereby a mixture of local and outside systems are interfaced to provide the revised ideal total integrated system. We are further along the way - we are not there yet. The development of automated acquisitions systems follows the model outlined. In the sixties, emphasis was on the mechanical ordering function and its attendant accounting needs; in the eighties, emphasis is on collection development, collection management, and rationalization. For the former needs, a stand-alone ordering/accounting system, whether manual or automated in some way, was probably sufficient. For the latter needs, all kinds of information derived from usage must be taken into account, requiring integration of the acquisitions function with circulation, catalogue information, and other functions. Management should no longer be looking at the process of acquiring items in isolation from other library processes, either in the decision-making or the mechanical stages.

From the beginning, with those batch, often outside-controlled systems of the sixties, little real development took place in the seventies as far as the utilities were concerned. There the focus was on bibliographic provision, with some service to locally based systems such as circulation modules, often via tape services. The acquisitions function was largely left to local ingenuity to develop, and partly this was due to the complexity and local variety of practices (see Gozzi, 1980). Only in the last three years or so have the utilities begun to offer solutions to acquisitions needs (see Madden, 1980). Meanwhile, libraries have attempted to share their local solutions with each other through the literature. Local usage ranges from Rice University's Fondren Library which has developed its fund-accounting and reporting on a word-processor in a situation where there is no other automated acquistions systems (Niles, 1983), to fully integrated multi-campus systems such as at Pennsylvania State University - there, LIAS (Library Information Access System), serving dispersed sites with a multi-function capability encompassing circulation, catalogue information, serials check-in and a future acquisitions/fund accounting among others, is well into its development (Cline, 1983). Northwestern University has outlined its NOTIS system (Furlong, 1980), and has now offered it to others (Northwestern to Offer NOTIS Services, 1983). In Canada, Mississauga Public Library's experience with UTLAS has been shared (MacIntosh, 1980), as has Memorial University's, also with UTLAS (Ellis and Straus, 1982). Several other large systems have been reviewed by Chen and Bressler (1982, p. 132-142). Some libraries have tried the bibliographic utilities' acquisitions modules and decided against them (see Calhoun and Bracken, 1982, for example) for economic reasons as well as for problems with complexity, response time, and meeting needs for access points and fund-control. A useful overview of two dozen locally developed or adapted systems was offered in the Oct. 1982 issue of Software Review (by Garoogian and Geffner).

> "There are three major, practical reasons for undertaking the automation of library functions: (1) to do something less expensively, more accurately, or more rapidly; (2) to do something which can no longer be done effectively in the manual system because of increased complexity or overwhelming volume of operations; and (3) to perform some function which cannot now be performed in the manual system - providing always that the adminstrator actually wants to perform the new service, has the resources to pay for it, and is not endangering the performance of existing services for which there is an established demand." (Veaner, 1970, pp. 300-301).

These comments apply very aptly to the acquisitions function. In addition to the three general reasons outlined by Veaner, Gozzi (1980, p. 167) points out that traditionally automation has been applied to the acquisitions function on three levels: (1) execution of specific clerical functions; (2) monitoring of the flow of materials through the system; (3) supplying assistance to management. Most early systems concentrated on the first two; today a shift in emphasis requires attention to the third. Reports may now be generated on such things as: titles costing more than X% of the encumbered estimate; vendor performance, delivery and discount; new acquisitions by subject/classification group; graphing of fund-accounting, etc. In the future, automated access to selection tools now redundantly scanned by collection development librarians may streamline the collection-building process (Kountz, 1980).

Given the above, how does a library go about deciding what kind of system to implement, especially if it has little in the way of existing automation of its functions? As Woods points out, "librarians should not run out and buy or develop a stand-alone acquisitions system without considering the advantages and disadvantages of compatible or integrated systems in the library." (p. 244). He further maintains that integrating the various functions can reduce the time and effort needed to develop new systems; the elimination or reduction of data duplication is another important advantage, since data duplication is not only costly but can also lead to a lack of data integrity. Integration also leads to better data management - not only can the same data be used by various users for various purposes, one person or group can be in charge of data base management. "Integrating a system creates an economy of scale so that more dollars are available to obtain the system desired", says Wood (1980, p. 245). Or does it?? Wood goes on:

> "A barrier to developing compatible systems is the expense. Available integrated systems technology have higher one-time costs, may require additional memory or a more powerful computer, and may result in more

> expense and time required for processing than is required for sequential processing. Integrated systems tend to be complex...the ability to recover from errors is more difficult because of the increased complexity and widespread use of the system. Worst of all, a failure in one component can stop the entire system." (Woods, 1980, pp. 245-246).

This raises the issue, then, of fully integrated systems versus compatibility of components, such as suggested earlier, whereby units of various configurations are joined and interfaced at points where they need to share/transfer data, but can work alone for manipulation of these data in performance of various local tasks. But as Woods points out, "the compatibility of systems in the long run hinges on standards...interfaces are the key to system compatibility." (Woods, 1980, p. 249). These interfaces are only now being developed to enable us to take advantage of this component/compatibility configuration. (See Stewart, 1982, for a brief report on standards for acquisitions / ordering systems, in particular those of BISAC, the book industry advisory group, aimed at accomodating order transmission between library purchasers and vendors in an automated mode - whether this interaction is direct from individual library to individual vendor, or through a utility, is one of the questions for libraries in the eighties — see Bonk, 1983; Bierman, 1980, and Aveney and Heinemann, 1983, for overviews of the developing automated vendor/library activity. This activity must be two-way, however, for full advantage to be taken of this possibility; vendors must be able to send status reports back to the library in the same automated way, and even invoice transmission and payment must be possible eventually. For vendor/utility co-operation, see "USBE and UTLAS, 1983 ; also Aveney and Heinemann, 1983, p. 45). UTLAS has ten or more vendors on-line with it now, available through both its ACCORD ordering module and its INNOVAC acquisitions module. Further discussion of library automation issues in general is beyond the scope of this paper, but the literature is plentiful (see, for example, Matthews, 1982; Matthews and Williams, 1983; Aveney and Heinemann, 1983; Gray, 1983). Some of the important issues to bear in mind include: start-up and on-going costs; the state of existing automation in the library and the question of integration with existing and future systems; availability of off-the-shelf software versus using an outside existing acquisitions module of a vendor or utility versus develop-your-own or adapt-someone-else's self-developed system; reliability of a supplier and experience of other users (ask around); response time (integrated systems may be slow because of the multiple demands being made on the same computing power, and peak periods of usage may have to be avoided); these are but a few of the vital issues.

As pointed out previously, libraries today have the option of choosing among a number of options for acquisitions. As Aveney and Heinemann avow,

> "Library acquisitions systems have, in fact, reached a relatively high state of the art. In general, there has been success in automating many of the traditionally

> labor-intensive processes in acquisitions. In automating these processes, foundations have been laid for new capabilities that build on the resources such as the large shared machine-readable catalogs and communications links which these systems put in place...The major theme in the automation of acquisitions and collection development is the drastic reduction of the human effort required for record-keeping and repetitive tasks so that greater portions of the acquisitions and collection development librarians' time can be spent doing what they do best: planning and deciding." (Aveney and Heinemann, 1983, p. 52).

Microcomputers and their use in libraries are a development of the latter part of the seventies;theycontinue to gallop ahead in capability, scope for use, and imaginative applications (see Krueger, 1981; Nash and Nash 1981; Stevens, 1983; Tayyeb, 1983; Dwyer, 1981; Hines [et al.], 1981, etc. for discussion of micros in libraries). In the early stages the uses, like those of the larger computers in the sixties, were mainly for local, small and unsophisticated systems. The early micros were themselves limited in their abilities - the vast majority of these functions were single-user, single-task situations. At the rate of current development, microcomputers with the power of the seventies' main-frames will soon become widely available. Software will also develop, and these machines will be capable of doing many of the functions, with integration of a number of areas quite nicely. They will be able to handle, at least for small and medium-sized libraries, the broad spectrum of automation needs, at a modest price. Even today though, micros still have limited capacity and software availability and the multiplicity of operating systems makes many packages incompatible with many machines. The shake-down of systems and the explosion of capability is on the horizon, however. The decade promises much to come in the way of change. To choose now is to know that in two years both the machine and the software will probably be obsolete - but this does not mean that you cannot keep using it. Five to seven years for replacement is a fair time-estimate, I think, and setting aside a portion of the expected replacement costs each year is probably less painful than trying to find it all at once when the time comes.

Software is the key to successful automation. In most cases it should be chosen before the hardware because it is much easier to fit the hardware to suitable software than to convert software written for one system to another. Software may be obtained in a wide variety of ways: you can buy off-the-shelf (see *Access: Microcomputers in Libraries*, vol. 2, no. 4 (Oct. 1982), p. 27, for announcements of two such packages), but these may or may not include fund accounting or allow for the volume of your operation, or the complexity of your budget structure, your large number of locations and copies, etc. You can buy such software and adapt it yourself (or hire someone to do this); or buy software developed by another library and adapt it; or you can develop your own; or contract for its development by a consultant. But even if you acquire acceptable software in some way, there is the question of on-going support/refinement/improvement of it in the future. Who can you turn to if trouble develops (e.g. your overlays get damaged by a power surge - how do you diagnose and correct such unexpected disasters)? And now, what machine do you want to run the software on? What operating system does it require? Have you a choice of machine or is it unique to one make? (For information on buying a micro, see Blair, 1981 etc.; Mason, 1983; for ergonomic factors, see Grandjean, 1980; Grandjean and Vigliani, 1980; Mason 1984). Very careful calculations must be made as to the expected size of files, programs, etc. in order to acquire sufficient storage and computing power for the job to be done, and for future expansion and needs. In many cases, if access/indexes are to be on-line, and financial analysis is an integrated part of the acquisitions system, then a rigid disk environment, where all files and records are available at all times, is probably best. Rigid disks now come with anywhere from five to sixty megabytes of storage, and response time is much better than for floppy systems. 64K of core memory is minimum; 128 or 256K is better. 16 bit architecture is rapidlybecoming the standard, although there are still many satisfactory 8-bit machines available - there is generally more software available for the 8-bit machines as they have been around longer. In the under $10,000 category there is little available with large storage capacity and multi-user capability - most micros are still for use on one task at a time. As a rule, software for micro acquisitions applications should be able to accomodate 5,000 records, have little limitation on record and file size, and should have generous field lengths. Record size should allow 500-1000 characters (less than 500 gives you very limited data capacity) and files should have large limits (several megabytes at least). Ideally, full-screen display (rather than line-by-line, field-by-field) means a lot more data can be displayed and seen on screen at a glance. A built-in report-generator is also desirable, as are multiple index points and Boolean search operators (AND, OR, NOT, =, etc.).

Chances are you will want to acquire a full data-base management system, which can be used for other applications, not just a pre-packaged acquisitions system. Or even better (for local customization), a program applications generator package. Data base management systems and their related software packages are designed to organize a large mass of data so you can get at it in many different ways without having to write a lot of complicated programs (for more information, see "Data Base Management Systems", 1982), while allowing local customization without requiring a lot of expertise on the part of the customizer or the user. Successful use of this type of software on micros, for various library functions, has been reported recently in the literature (see Cole and Hill, 1982; Chan, 1983; Sukovich, 1983). DBase II has proved popular as a basis for acquisitions systems. Others have developed their own micro systems in-house (Walker, 1982). As pointed out in Small Computers in Libraries, (Feb., 1982, p. 5), turnkey systems are also possible, since there are some "single-function" programs available for the smaller personal micros. They are useable by small libraries with limited and simple needs but their capacity is limited to a small number of records (in the hundreds or low thousands). Once you wander into the "total systems" approach you are no longer talking modest dollars/modest capacity - you are in the world of multiple users and well out of the "do-it-yourself" range.

A library can only decide what to do after extensive analysis of the function(s) to be performed, current/future possibilities both in-house and elsewhere, and much consideration of the possibilities; many of the factors have already been alluded to in this paper. So, taking all the above into consideration, let us follow one library's decision-making process to its conclusion.

The R.W.B. Jackson Library of the Ontario Institute for Studies in Education is the subject-focused centre of an autonomous academic organization charged with the tripartite functions of educational research, dissemination and implementation of the research, and the provision of graduate-level instruction in education. For the latter function it is voluntarily affiliated with the University of Toronto through which are granted Masters and Doctoral degrees; in addition, the Institute conducts diploma-level courses in some areas of its work. The library supports all three functions; it holds upwards of 500,000 items, both print and non-print, and adds about 8,000 titles a year to its collection, placing approximately 6,000 individual orders per year (excluding subscriptions). We have twenty-one different locations or sub-collections on our three floors, representing sixteen different budget allocations; we currently collect thirteen different categories or types of materials, ranging from books to audio-visual items and materials for school use including computer programs on disks. The above detail should give you some idea of the complexity of our acquisitions framework in spite of our relatively modest volume of titles ordered per year.

While the library has of course offered sophisticated on-line information searching to its patrons for a number of years, we have not yet automated our internal tasks to any great extent, other than our searching/cataloguing functions. Since April 1978 we have been members of the UTLAS Network, and of the University of Toronto Campus Union Catalogue Automated Users Group. We handle all types of materials and cataloguing records through UTLAS - if we catalogue it, the record goes into the UTLAS data base; monographs, serial titles, and all non-print materials are included. Since June 1980 we have been automated authority users as well. Public Services shares our terminal and uses REFCATSS, mainly for interlibrary loan searching. Except for our AV records we have not been able to do any retrospective conversion of the 100,000 cataloguing records created prior to 1978; for our AV collection we do produce a COM catalogue twice a year of the complete holdings in this area. Otherwise we are living with our existing card catalogues, the cards being created by UTLAS from our records in the data base. We have no automation in our Circulation function (we have only seven years of machine-readable records - about 30,000) and estimate that it would cost about $500,000 to upgrade to AACR2 and enter into the UTLAS data base - original cataloguing rates are relatively high with us (30% or so, and the older material will be even higher). We are considering options for conversion and for freezing the card catalogues but have no firm plans for the foreseeable future. Circulation is still in our minds.

In the Autumn of 1980, after implementation of UTLAS CATSS (1978) and authorities (1980), the senior author of this paper began looking at acquisitions automation possibilities. Something had to be done in the light of shrinking staffing and other complexities. It was determined that the integrated model would theoretically be best; if the library was going to be able to automate other functions, such as circulation, it would be better to look at various automation needs and consider them together. After some investigation and a look at what was available at that time, it soon became apparent that there simply was not the money to consider full retrospective conversion of the 100,000 cataloguing records, acquire hardware and software to run a Circulation system, and at the same time have a fully integrated acquisitions module as part of the package - and serials control was not even a reality at this point!

After a pause to explore possibilities of financing with our management, the only alternative that seemed feasible economically was to go ahead with an acquisitions module on its own, as that is where our need seemed most serious and the costs most likely to be supportable. By now it was well into 1981, and our institution was starting to look at its computing needs for the future for all its divisions. Mary became a member of the advisory committee looking at needs and solutions, and placed the library's case before it. Reality dictated that the library's "slice-of-the-pie" available for automation would likely be enough only for acquisitions. But via what method?

UTLAS, our utility, instituted its ACCORD ordering system just about the time we were seriously considering alternatives. Using this facility we could print orders locally, have access to on-line vendors, and key our data only once. There was, however, no fund accounting provision with this option. During this time, also, UTLAS introduced its INNOVAC acquisitions system, based on a locally held microcomputer and including fund-accounting, with full interface to the UTLAS system for data-transfer etc. While ACCORD is pay-as-you-go after set-up charges, INNOVAC requires a purchase fee plus maintenance monthly. We seriously looked at both these solutions, given the advantages which the literature recommended, e.g. interfacing and compatibility. But we also looked at stand-alone microcomputers as well.

By now it was the Summer of 1982; in the Autumn of that year our institution made a general committment to the product line of Xerox, for both microcomputers and word processing; models to be purchased were the 820-II microcomputer and the 860 word processor. By then we knew that the estimated costs of ACCORD ($21,000 per year) and INNOVAC ($70,000 purchase, $500 per month maintenance) were beyond our means, even if they offered all the advantages of integration/compatibility with the huge UTLAS data base, one-time keying of data, etc. etc. For us , taking advantage of either of the modules offered by our utility was not feasible or justifiable, nor did they really meet all our defined needs in details wanted in such a system, given our complexity. It thus came down to a micro plus software, a package we felt we could acquire for about $14,000 total, or nothing-at-all. And nothing-at-all was not a viable answer since we now knew that by May 1, 1983, the start of our new budget year, we would be reduced from four clerks to three in the order function, and possibly to two at some time during the 83/84 budget year.

So in Nov. 1982 we made the committment to the stand-alone micro route, and also accepted the micro model that others were going to acquire elsewhere in our organization, namely the Xerox 820-II. Given the estimate of the size of a year's operation, we knew we needed a ten-megabyte rigid disk drive to accomodate a core-file of several thousand vendors as well as several thousand order records and their indexes, the report information needed, and all necessary programs. A top-quality dot-matrix printer would suffice since speed was more important than a letter-quality finish on our purchase orders and financial reports.

But what software would we get? There was no pre-packaged system that would handle our complexity and volume, at least that we could find. The 820-II is a CP/M based Z80A processor with 64K of memory, a 24 line by 80 character display of white on black with two serial and two parallel I/O ports, available with either two floppy drives or a rigid drive with one floppy slot (8", double-sided, double-density, for 980K of storage on each floppy). The ten megabyte rigid not only gave us room for all our files at once, the high-density disks made backing-up our data files easier since the operator does not have to change disks so often.

Well, we needed CP/M software. It was evident that what we wanted was a basic data-base management system package that could be customized to our needs. It must provide access by author, title, ISBN and purchase order number on-line, and be capable of retrieval on other fields in a batch mode as required in special requests (our old system was accessible by exact title only). It must also provide for batch financial reporting, monthly (on-line was not considered necessary), without extra keying-in of the financial data - in other words, from the order record itself. As mentioned, special reports must be possible. To cut down our monthly accounting from seven clerical days to hours seemed vital in light of our staffing realities. Automatic generation of purchase orders in daily batches was also required; keying of vendors only once was another given - ability to pull in data from a permanent vendor file to the transient purchase order file and use data from both files in generating purchase orders was also needed. No envelopes need be typed since we would use window envelopes and the vendor name and address on the p.o. form would suffice for mailing. Since we use thousands of vendors in a year (we had 30,000 in our manual address file), and generally do not use jobbers, our daily volume of orders goes to many different addresses, so an individual order-form for each title would be appropriate. (Continuous feed, three-part order forms were custom-designed jointly by Buddy and the representative of the supplier of the forms).

But who would do the custom-work for us on our software, and using what package? We realized that this was happening backwards - we "chose" the equipment before having software to run on it, contrary to what all the wisdom advised. But given the situation of being swept along in the movement of our parent organization, and in the interests of equipment compatibility (not a bad idea if breakdowns occur since you can borrow pieces, use others' identical models in a crisis, and even take disks to other machines to gain the use of a letter-quality printer, etc.) and volume discounts on purchases, we decided to accept the recommended machine.

Again, contrary to the wisdom of the literature, we did things the "wrong-way-round" and accepted the advice of our hardware salesman - we hired a relatively new firm, Micro Software Sales, in the person of my collaborator, Buddy Doll, to provide the customization. By now it was January of 1983 (these things do not happen quickly!); our deadline for operation was May 1, 1983 - we must be up and running in perfected condition by that date when we would start a new budget year and lose a clerk.

And that is when, how and why Buddy Doll and Mary Campbell joined forces to together produce yet another customized stand-alone order/accounting system, against all the growing advice of the literature. And yet, for our library, it was the only solution. Starting with a basic data-base package and a list of required fields and features, we worked for several days each month through January to April, implementing, testing, refining, revising, improving, correcting, and debugging, occasionally making compromises but achieving a product rather close to our wants. We used a carefully constructed set of test records (about 80) designed to reflect all the complexities of our structure and to simulate our reporting needs. Believe it or not, we had it ready three weeks ahead of time! We even loaded hundreds of vendors prior to start-up day.

Commencing on May 1, 1983, starting with new orders placed as of that day, we have gradually geared up the new system while running the old one for items already on order before that day and carried over from the previous budget year. A year into the project, by far the majority of transactions now are on the micro system. We handle all orders (excepting subscriptions) on it, for all types of materials, even standing orders, and most accounting through it as well.

For various reasons, including an unexpected doubling of the acquisitions budget due to the special one-time grant to university libraries, it became clear to us by Autumn 1983 that although the system itself worked well, the software had various limitations inherent in it, such as file sizes, that rendered it unable to handle the unpredicted increase in volume. So we cast about for a new basic package to re-adapt to our new needs.

And that is when Q-PRO 4 came into our lives. We again hired Micro Software who recommended it and again did the customization. Q-PRO 4 is a fourth-generation application development system for microcomputers (see Heintz, 1983; Stewart, G., 1982, for discussion of this type of program generator).

Program generators are problem-oriented rather than procedure-oriented. You can custom-tell your computer exactly what to do without having to learn a programming language. Screens can be entirely customized, in full-screen mode, exactly as you like; fields may be any length you wish and these can be changed and re-defined at later stages very easily - screen design, data file description, and program procedures are all easily done in a much shorter time than is original programming. Your programming needs can be addressed in layman's terms; conversion to technical terms is handled by the package itself, but with greater flexibility on the part of the user. These packages also contain reporting capabilities and mathematical manipulation features.

To use a program generator effectively you must fully understand the desired application. It's no use trying to make the program generator produce a complicated ordering and fund/accounting system if you do not have a minute grasp of all the details and stages that go to produce it. To design such a system requires a complete systems analysis and charting to determine the exact needs of your data for storage (how many characters do you need for each field? What fields do you need? etc. etc.). We were lucky in that we had done this work for our original software, and had a very clear idea of how we wanted our system configured. Design of the new software commenced in Dec. 1983 and implementation took place in Feb. 1984. We were constrained only by needing to conform our output to match the existing multiple order forms designed for our first system and to bear in mind the data file already created which had to be converted to the new format. Conversion was done by a specially written program, run over a week-end after two trial runs - we were out of commission for a few days but quickly recovered and, after correcting two small bugs, are happy with the results - we have an even better system with a much larger potential.

Q-PRO 4 has no restrictions on such items as record size or number of fields and allows generous field lengths. Files may be up to eight megabytes in size and up to 255 files may be open at once. It handles index sequential, random and sequential files. The programmer is free to set up the data base in any way, heirarchical, relational, etc. The two most time-consuming and tedious tasks for a programmer, user interface and reporting, are simple, fast and straightforward. Formatted, fill-in-the blanks screens are implicit in field editing. The programmer has complete control over entry and handling of all data entry fields. Highlighting and screen graphics are available if the CRT supports them. Q-PRO 4 is designed to run on Z80 or 8086/8088-based microsystems and is available for a variety of operating systems (CP/M, MP/M, MMMOST, MS-DOS, PCDOS, TURBODOS). It runs on such micros as Intertec, Advantage, various Televideo models, Zenith, Vector Graphic, TRS 80-II, Osborne I, Apple, Xerox 820, Adm 3A, DEC VT100, ADDS Viewpoint, Epson, Hazeltine 1500, Kaypro, IBM-PC, Eagle and others. The 8 bit version requires 64K of memory, the 16 bit version needs 128K. Q-PRO 4 is available in either single or multi-user versions. In most ways Q-PRO 4 meets or exceeds the requirements for a data base management system as defined in the literature, and is extremely versatile for custom applications. Prices are competitive with comparable packages, ranging from $395 for the 8 bit single user version to $795 for the 16 bit multiple user version. Customization is, of course, extra, or you can teach yourself.

So what have we given up, and what have we gained by being forced into a stand-alone acquisitions system? We do not have any communication with our utility - so we do end up with double-data entry for some fields; but since we use only eight of the bibliographical fields in our order records (the rest of the order record is ordering/accounting information) and don't need the others, the duplication is not all that great. And since approximately ten percent of our orders are never filled, and others have changes required to the original data when received, a significant amount of our data would have to altered anyway or deleted. This way we are not paying for data of temporary value. In addition, we have complete flexibility and customization to our particular internal design and structure with regard to budgets, locations and types of materials. No compromises of input or output are required in order to fit into a predesigned system. What we don't have is on-order integration with items already in the collection - other users of UTLAS get no warning of our intention to acquire an item, sometimes useful in a shared environment. And we have to search two places before placing an order and to check on an item's progress or existence. However, our micro system gives us a "track" of every item in-process (outstanding, received, accepted and sent for cataloguing, finished and in the library) that we wouldn't have in other systems. We do not have direct vendor access, but since the ten or so now in the UTLAS system are not likely to be used often by us, given our choice to use a wide variety of suppliers, this is not a big loss to us. We also don't have the capability of accepting other UTLAS developments of local modules for the INNOVAC such as the recent serials module since the INNOVAC operating system is unique to that microcomputer. But in addition to the INNOVAC, UTLAS has recently announced (memo, Feb. 28, 1984) that it intends to adopt the IBM PC as its micro-of-choice for use as an UTLAS terminal when it changes from a 300 BAUD service to 1200 BAUD later in 1984. Local systems may also be developed for this equipment in the future, perhaps at more affordable rates; if we purchase this equipment to use as our regular communication to UTLAS, a whole new set of possibilities will open up to us.

Meantime, we are contentedly running our immediate solution to our problem, our stand-alone, recently re-invented, customized "wheel", which, while looking for all intents and purposes like an unwise move theoretically, is the only way we could get started. For us, it serves the purposes, for the time being, in a very satisfactory way.

For a look at our main order-record screen, please see below (we also have a main menu, a vendor-record screen, a key/index searching screen, a history-fields screen, an order-printing screen, and several financial-reporting screens). Our four main files are: vendor data; order data; keys/indexes; and financial data (created temporarily and automatically from the financial fields in the order record, for each run of the financial analysis). As of March 1984 the vendor file occupies 400,000 bytes with about 1400 suppliers in it so far - we intend to build it to several thousand and 2 or 3 megabytes in size; our order data file stands at 2.5 megabytes, representing 4,000 orders; and our key-files (indexes to the order records) stand at 1.25 megabytes - already we are using about half our disk storage space. At the end of our budget year archival disks will be taken and all finished records deleted from the active data base, freeing a good portion of the disk space to be used again in the next budget year. Response time will also improve at that point. This is a current, interactive data base, accessible by four fields in the order record and one in the vendor record; it is very much used. In fact, it is in use eight or nine hours a day by staff doing various tasks on various parts of the system, and our next move is to explore acquiring hardware capable of multi-tasking/multi-user activity, since we are rapidly exceeding single-user staff time available with our extra heavy ordering level. It would also enable us to explore adding other functions such as serials control and provide direct access to the on-order/in-process file for Public Services through a read-only query terminal in their area which is on a different floor than the Order facility. With this in mind, we bought and developed the multi-user version of Q-PRO 4 although we are running it in a single-user mode at the moment. Since it will run on a variety of machines our future options are several and open.

In sum, the decision was right for us at the time it was made and will continue to serve us for the near future at least.

REFERENCES

"Automated Acquisitions Systems: Papers Presented at the LITA Institute, Part I [and] Part II". Journal of Library Automation. Vol. 13, no. 3, (Sept. 1980),pp. 155-195; Vol. 13, no. 4, (Dec. 1980), pp. 221-264.

AVENEY, Brian and HEINEMANN, Luba. "Acquisitions and Collection Development Automation: Future Directions". Library Hi Tech. Vol. 1, no. 1, (Summer 1983), pp. 45-53.

BEAUMONT, Jane and KRUEGER, Donald, eds. Microcomputers for Libraries: How Useful Are They? Ottawa: Canadian Library Association, c1983.

BIERMAN, Kenneth J. "Vendor Systems and On-Line Ordering". Journal of Library Automation. Vol. 13, no. 3, (Sept. 1980), pp. 170-181.

BLAIR, John C., Jr. "Micro Magic". Online. Vol. 5, no. 4, (Oct. 1981), pp. 90-94; vol. 6, no. 2, (March 1982), pp. 64-67; etc.

BONK, Sharon C. "Integrating Library and Book Trade Automation". Information Technology and Libraries. vol. 2, no. 1, (March 1983), pp. 18-25.

BOSS, Richard W. "Automated Acquisitions Systems: Keynote Address". Journal of Library Automation. vol. 13, no. 3, (Sept. 1980), pp. 156-163.

MICRO BASED ACQUISITIONS

BOSS, Richard W. Automating Library Acquisitions: Issues and Outlooks. White Plains, N.Y.: Knowledge Industry Publications, 1982.

CALHOUN, John C. and BRACKEN, James K. "Automated Acquisitions and Collection Development in the Knox College Library". Information Technology and Libraries. vol 1, no. 3, (Sept. 1982), pp. 246-256.

CHAN, Anna. "dBase II in a Small Library". Small Computers in Libraries. vol. 3, no. 8, (Aug. 1983), pp. 4-5.

CHEN, Ching-Chih and BRESSIER, Stacey E., eds. Microcomputers in Libraries. New York: Neal-Schuman Publications, c1982.

CLINE, Nancy. "LIAS - Library Automation With An Integrated Design". Library Hi Tech. vol. 1, no. 2, (Fall 1983), pp. 33-48.

COLE, Susie and HILL, Cindy. "Automating Library Systems with PFS". ACCESS: Microcomputers in Libraries. vol 2, no. 4, (Oct. 1982), pp. 11-14.

"Data Bases Management Systems". Small Computers in Libraries. vol. 2, no. 3, (March 1982), pp. 2-5.

DE GENNARO, Richard. "Library Automation & Networking: Perspectives on Three Decades". Library Journal. vol 108, no. 7, (April 1, 1983), pp. 629-635.

DWYER, James R. "Micros: Applications Tomorrow". ACCESS: Microcomputers in Libraries. vol. 1, no. 2, (Oct. 1981), pp. 3+.

ELLIS, Richard and STRAUS, Leslie. "Electronic Ordering at UTLAS: A Chronicle of Library/Book Vendor/Bibliographic Utility Cooperation". Information Technology and Libraries, vol. 1, no. 4, (Dec. 1982), pp. 343-344.

FURLONG, Elizabeth J. "A Case Study in Automated Acquisitions: Northwestern University Library". Journal of Library Automation. vol. 13, no. 4, (Dec. 1980), pp. 222-240.

GAROOGIAN, Rhoda and GEFFNER, Bonnie. "Computer Software for Libraries and Information Centers". Software Review. vol. 1, no. 2, (Oct. 1982), pp. 124-144.

GOZZI, Cynthia I. "Local Systems: Design and Costs". Journal of Library Automation. vol 13, no. 3, (Sept. 1980), pp. 164-169.

GRANDJEAN, Etienne. Fitting the Task to the Man: An Ergonomic Approach. England: Taylor & Francis Ltd., 1980.

____ and VIGLIANI, E., eds. Ergonomic Aspects of Visual Display Terminals. England: Taylor & Francis Ltd., 1980.

MICRO BASED ACQUISITIONS

GRAY, Carolyn M. "Turnkey Systems: Selection and Installation". Library Hi Tech. vol. 1, no. 1, (Summer 1983), pp. 59-61.

HEINTZ, Carl. "Generating Program Applications the Quic-N-Easi Way". Interface Age. vol. 8, no. 10, (Oct. 1983), pp. 116-118.

HINES, Theodore C. [et al.] "Micros: Applications Today". ACCESS: Microcomputers in Libraries. vol. 1, no. 2, (Oct. 1981), pp. 2+.

KOUNTZ, John. "Automated Acquisitions Systems: A Survey". Journal of Library Automation. vol. 13, no. 4, (Dec. 1980), pp. 250-261.

KRUEGER, Donald R. "Issues and Applications of Microcomputers for Libraries". Canadian Library Journal. vol. 38, no. 5, (Oct. 1981), pp. 281-285.

MACINTOSH, Helen. "The Mississauga Library System: A Case Study". Journal of Library Automation. vol 13, no. 4, (Dec. 1980), pp. 240-243.

MADDEN, Mary A. "The Role of the Network in Automated Acquisitions". Journal of Library Automation. vol. 13, no. 3, (Sept. 1980), pp. 181-185.

MASON, Robert M. "Choosing a Vendor: Where to Buy a Micro". Library Journal. vol. 108, no. 6, (March 15, 1983), pp. 560-562.

____ "Ergonomics: the Human and the Machine". Library Journal. vol. 109, no. 3, (Feb. 15, 1984), pp. 331-332.

MATTHEWS, Joseph R. "20 Qs & As on Automated Integrated Library Systems: Are They a Remedy for What AILS Your Library?" American Libraries. vol. 13, no. 6, (June 1982), pp. 367-371.

____ and WILLIAMS, Joan Frye. "Oh, If I'd Only Known: Ten Things You Can Do Today to Prepare for Library Automation Tomorrow". American Libraries. vol. 14, no. 6, (June 1983), pp. 408-412.

MCCUNN, Donald H. "Evaluating Equipment: Criteria for Word Processing Hardware". ACCESS: Microcomputers in Libraries. vol. 2, no. 3, (July 1982), pp. 14-16.

NASH, John C. and NASH, Mary M. "Libraries and Small Computers - Perspective for Decision-Making". Canadian Library Journal. vol. 38, no. 4, (Aug. 1981),]]/ 207-211.

NILES, Judith. "Acquisitions Fund Accounting on a Word Processor". Library Hi Tech. vol. 1, no. 2, (Fall 1983), pp. 10-13.

"Northwestern to Offer NOTIS Services". Information Technology and Libraries. vol. 2, no. 3, (Sept. 1983), pp. 326-327.

Q-PRO 4: the Fourth Generation Applications Development System for Microcomputers. Weston, Ont.: Harso Ltd., 1983. [4 p.]

Q-PRO 4 [User's Manual]. Langhorne, Pa.: Quic-N-Easi Products, Inc., 1983.

ROSENBAUM, Linda. Health Effects of Video Display Terminals: the Non-radiation Problems. Toronto: Health Advocacy Unit, Dept. of Public Health, City of Toronto, 1981.

SCHRAME, Mary L. "The Psychological Impact of Automation on Library and Office-Workers". Special Libraries. vol. 72, no. 2, (April 1981), pp. 149-156.

STEVENS, Norman D. "Hi(gh) Tech(xpectations)". Library Hi Tech. vol. 1, no. 1, (Summer 1983), pp. 89-90.

STEWART, Charles C. "Update on Ordering Standards". Information Technology and Libraries. vol. 1, no. 4, (Dec. 1982), pp. 341-343.

STEWART, George. "Program Generators". BYTE. vol 7, no. 8, (Aug. 1982), pp. 38-56.

SUKOVICH, John E. "Acquisitions with dBASE II". Small Computers in Libraries. vol. 3, no. 8, (Aug. 1983), p. 7.

TAYYEB, Rashid. "Microcomputers: Implications for Technical Services". Canadian Library Journal. vol. 40, no. 2, (April 1983), p. 64.

"Turnkey Trends". Small Computers in Libraries. vol. 2, no. 2, (Feb. 1982), p. 5.

UDEN, Janet. "Financial Reporting and Vendor Performance: A Case Study". Journal of Library Automation. vol. 13, no. 3, (Sept. 1980), pp. 185-195.

"USBE and UTLAS". Information Technology and Libraries. vol. 2, no. 3, (Sept. 1983), pp. 327-328.

VEANER, Allen B. "Major Decision Points in Library Automation". College & Research Libraries. vol. 31, (Sept. 1970), pp. 299-312.

____ "Technical Services Research Needs for the 1990s". Library Resources and Technical Services. vol. 27, no. 2, (April/June 1983), pp. 199-210.

WALKER, Stephen. "A Microcomputer Based Library Acquisitions System". Small Computers in Libraries. vol. 2, no. 8, (Aug. 1982), p. 6.

WOODS, Richard. "The Compatibility of Library Systems". Journal of Library Automation. vol. 13, no. 4, (Dec. 1980), pp. 244-250.

WOODSON, Wesley E. Human Factors Design Handbook: Information & Guidelines for the Design of Systems, Facilities, Equipment & Products for Human Use. McGraw-Hill, 1981.

```
                              ORDER RECORD
 Order Number: [19-1602-000]    Title Key: [EGERRYER]    Author Key: [MCDONEIL]
Vendor Number: [00000132]  [Gage Publishing Ltd.                   ]
        Title: [Egerton Ryerson and his times.                     ]
               [                                                   ]
       Author: [McDonald, Neil; Alf Chaiton.                       ]

ISBN/LCCN: [0770517064] Yr Pub.: [1978]  Edition: [    ]  Volumes: [            ]
Number of Copies: [   2] Place: [Toronto      ] Publisher: [Macmillan of Canada ]
*-------------------------------- DATES -----------------------------------*
      CREATED                RECEIVED              ACCEPTED              FINISHED
     [83/10/28]             [83/11/10]            [83/11/14]            [83/11/14]
*-----------------------------------------------------------------------*
Unit List Price      Total List Price      Total Cost Price      Type Of Material
 [      8.00]         [      16.00]         [      32.73]          [01]  [BK]

Budget Allocation: [07]  [COURSE      ]      Library Location: [08]  [CIRC      ]

Rush: [RUSH]  Rush Date: [83/10/28]     Cancel: [ ]       Course Number: [1414   ]
          Special Instructions                                    Series
[PLUS 379.71:R993:E29:CIRC           ]     [                                     ]
Claim Status: [                                            ] Standing Order: [ ]
 H1 PRINT SCREEN    H2 NEXT RECORD    H3 PREVIOUS RECORD    H5 RETURN TO MAIN MENU
  H6 HISTORY SCREEN    H8 KEY SCREEN    H9 REDISPLAY    H0 DELETE RECORD
```

ONT. INST. STUD. EDUC.
LIBRARY 3RD FLOOR
252 BLOOR ST. WEST
TORONTO, ONTARIO
CANADA M5S 1V6

DATE 83-10-28

RECORD NUMBER 19-1602-000

SPECIAL INSTRUCTIONS PLUS 379.71:R993:E29:CIRC

RUSH

TITLE / AUTHOR (SERIES)

Egerton Ryerson and his times.

McDonald, Neil; Alf Chaiton.

EDITION

YEAR PUB. 1978

ISBN / L.C. 0770517064

NO. OF COPIES 2

VOLUMES

COURSE NUMBER 1414

LIST PRICE $16.00

LIB. LOCATION CIRC

VENDOR

Gage Publishing Ltd.
164 Commander Blvd.
Agincourt, Ont.
M1S 3C7

PUBLISHER Macmillan of Canada
ACCOUNT NO. 80610

PLACE Toronto

VENDOR NOTES

LIBRARY PURCHASE ORDER

BEST COPY AVAILABLE

139

140

JUSTIFYING ONE-TIME USE PROGRAMS FOR SPECIAL DATABASE APPLICATIONS

John C. Nash
Faculty of Administration
University of Ottawa
Ottawa, Ontario, K1N 9B5

Mary M. Nash
Nash Information Services Inc.
1975 Bel Air Drive
Ottawa, Ontario, K2C 0X1

ABSTRACT

Conventional techniques for building and using textual databases focus on adapting the data to an established software package or depend on the orderly and well-documented construction of special software. The purpose of this paper is to illustrate how in some instances it may be acceptable practice to prepare disposable programs to handle special textual databases. This approach will be illustrated with the problem of cataloguing a reasonably large collection of graphical materials under severe time constraints, using a limited capacity microcomputer. The implementation decisions and management choices which were made will be discussed. Some of the building blocks which were used for the disposable programs will be noted. An attempt will also be made to compare this approach with more usual ones.

L'utilisation de logiciels jetables pour certaines bases de données particulières

---

Les techniques conventionnelles de conception et d'exploitation des bases de données consistent soit dans l'adaptation des données à un progiciel existant, soit dans la rédaction méthodique et documentée d'un logiciel particulier. Le présent exposé a pour but de montrer comment il pourrait être avantageux, parfois, de préparer des logiciels jetables pour exploiter certaines bases de données textuelles. Pour illustrer cette approche, on analyse un problème de catalogage, sur un microordinateur à capacité limitée, d'une assez importante collection de documents graphiques et ce, avec une sévère contrainte de temps. L'on discute également les choix administratifs qui furent faits, de même que les modalités d'applications du modèle. L'on présente quelques-uns des modules utilisés pour la préparation des logiciels jetables. Enfin, on compare cette approche avec des méthodes plus traditionnelles.

1. INTRODUCTION

This paper is concerned with a problem in information management where the underlying task was the creation of a catalogue of audio-visual material. The particular problem was special, and the computer techniques we used to solve it were unique in their actual details. We consider the programs to be purpose-built tools which were discarded afterwards. However,the ideas behind the programs have quite general applicability to information management tasks where the raw data is organized into ensembles or databases involving varying sizes of records as well as varying numbers and sizes of fields. In the present case, the fields used to index this collection may vary in number from zero to three, and the number of terms in a single index field may vary from one to five.

The project giving rise to this problem stemmed from the need to organize a collection of photographs, negatives, slides, films and videotapes. After some of the material was discarded, 7893 items were included in some 1520 records in the final catalogue, which was produced in loose-leaf form consisting of an accessions list and three indices -- a subject index, a geographic index and a personality index. The indices record what, where and who the photographic material concerns. The entries in the accessions list describe the materials by date of origin, caption or title (if any), size, medium, format and colour, and the subjects, geographic location, or personality indexed. Some catalogue (i.e. database) records may be simply cross-references e.g. from photographs to their corresponding negatives. The necessity for these records arose from the way in which the collection was housed, with separate storage for the different media; that is, it is physically inconvenient to intermix negatives and slides in storage. Rather than duplicate the descriptive content for some catalogue records, we used cross-references.

We were given approximately 50 working days to complete the entire cataloguing project. Of this time period, 34 days were spent actually coding and processing the material (including labelling and arranging storage). Though this gives an average of approximately 45 records created per day, the most productive day yielded 77 new records (and a very tired cataloguer).

Thus the project to build the catalogue involved the triple difficulties of:

- wide variability of record structure and size
- media which must be stored separately but which are logically related by content
- time pressure for completion of the project.

2. POSSIBLE APPROACHES

In considering how to carry out this project, we had the following alternatives:

a) a purely manual method based on index cards;

b) a microcomputer-based method using commercially available software;

c) a microcomputer-based method using self-programmed software.

Other possibilities, such as the use of a minicomputer or mainframe, were not available to us.

Of the three alternatives, only (c) was considered practical. The purely manual approach, while perfectly straightforward and feasible, did not lend itself to development of the three indices (subject, geographic, personality), since it would require multiple copies of each catalogue card. Furthermore, any typographical errors would be awkward to purge.

Commercial microcomputer software suitable for building catalogues and indices is primarily of the database management variety. At the time the project was undertaken (January to March 1983), we knew of no software explicitly intended for cataloguing. Some programs have recently been advertised, in particular in the Library Association Record (November 1983, vol.85, no.11, page 416); however, we have no personal experience of these offerings. Of the database management packages, Ashton-Tate's dBASE II is one of the better known and serves for illustration.

The first consideration for use of any package is its capacity to handle the amount of data involved. Many database packages, and dBASE II in particular, use fixed field sizes and fixed number of fields per record, hence fixed record sizes. While the fields may be left empty, the empty space is still allocated on disks and in memory. The biggest fields and the records with the most fields therefore determine the overall memory and storage requirements. In our case, the biggest record has approximately 2000 characters. This exceeds the maximum for dBASE II, although techniques involving associated databases could be used to overcome this difficulty. However, the total space requirement with a package like dBASE II is still much higher than with programs using variable length records. The database packages also need custom programming to produce suitably formatted printouts of the catalogue and indices. Therefore, even if the primary software is available "off-the-shelf", it requires:

1. adaptation of the data to accomplish the requirements of the database software;

2. programming of the commands to prepare printed catalogues and indices.

Additionally, in our own case, there was the need of acquiring, installing and learning a database package.

3. THE INDEXING AND CLASSIFICATION SCHEME

After examining the indexing systems and classification schemes of various other audio-visual collections in Ottawa, we decided to treat the material according to a modified AACR II scheme.

The unique element in each record is the accession number. This is an alphanumeric code consisting of a letter, indicating the type of material

N = negative, P = photograph, S = slide, F = film, V = videotape,

followed by the year that the item was created, followed by a number showing the order in which the catalogue entry was created for that year. For example, S82-48 is the 48'th item that we catalogued in the "slide" category for the year of origin 1982.

Each record also contained one or more subject headings. These were modified LC subject headings, initially modified by a research institute in Canada and further altered by us to reflect current practice in the subject field. There were (eventually) a total of 51 subject headings to choose from. The subject headings were cross-referenced with "see" and "see also" references. Another index reflected the possible 82 geographic locations where the material may have originated. The personality index had 105 different personalities listed. The complete description of any item (or group of items) was only given in the accessions list.

4. SOFTWARE USED

The main software functions involved in this project were:

- entry of the catalogue accession records;

- printing of the catalogue accessions list in a reasonable order;

- assembly of the index entries in alphabetic order;

- printing of the index entries.

Because the classification and structure of the collection was highly specialized, we decided to use programs which could be discarded after the project was completed. Our task was to produce a catalogue, NOT generate cataloguing software. Therefore, we could take the unusual steps, from a programming point of view, of

- ignoring program efficiency in favour of correctness of results and savings of human time to complete the project;

- omission of checks on data and user-friendly features which required extensive program code;

- no documentation whatever outside of program comments used in debugging.

This strategy was quite successful in this particular project. However, upon learning that we intended to use a computer to carry out the work, our client anticipated that he would be able to take over the database and programs, thus having an automated system made-to-measure to his application. Given the ensemble of undocumented programs we had used to perform the principal task and to patch errors or overcome unforeseen difficulties, we were in no position to supply software to users.

Our strategy does not imply that we ignored all user-friendly features. In the data-entry program, a number of checks and aids to detection and correction of errors were especially included, since accuracy at this stage in the work was critical to the quality of the product. Furthermore we employed several programs to check or print the contents of files. These checks were supplemented by careful use of a simple data-entry sheet for each record and a set of index cards, one (or more) for each year and type of material, in order to tally the accession numbers so that we could avoid duplication or non-use of accession numbers. Also the data-entry sheets could be used to record progress, an important consideration since there was a deadline to meet.

In correcting entries and in initially preparing the index terms we made use of word processing software of our own creation along with an editor from Anderson Technoproducts Inc. of Ottawa.

5. STRUCTURE OF THE SOFTWARE USED

The starting point for the software generation was the establishment of the initial set of indices. These had records of the form

```
   (code) (delimiter) (index term)
```

or

```
   - see (index term)
```

or

```
   - see also (index term)
```

The delimiter chosen was "~" (ASCII character number 126). Codes were a number prefaced by S, G, or P for subject, geographic or personality indices respectively. The indices were in separate files on a diskette. The numbers had no real meaning, since the indices were stored alphabetically by index term. An actual sort can be avoided, since we can enter each index record at the proper position using a text editor. Moreover, the indices were not built all at one time. An initial block of terms for each index was created based on a preliminary inspection of the materials and discussions with the client. However, as the cataloguing proceeded, new geographic locations and new personalities as well as new subjects were encountered. The only operational nuisance we encountered in entering the index terms was the need to reformat the internal storage of the files due to a peculiarity of our computer system. However, the reformatting permitted very much more rapid reading of the indices by our programs, which were coded in BASIC on a North Star Horizon. Note that a simple sorting program cannot be used to sort the index files alphabetically because of the "see" and "see also" entries.

Having numbered index terms greatly speeded up the coding and data entry, but required that completed entries be checked very carefully. By providing a printoff for completed entries, and using a numerically ordered listing of the indices, we could correct catalogue accession entries relatively easily. Here a sorter can produce such numerically ordered lists if the "see" and "see also" lines are dropped from the alphabetically ordered file. Two points we noted in dealing with the index files were:

1. Index terms longer than one line (approximately 64 characters) caused programming inconvenience and cataloguer stress, since they were not easily displayed and distracted the eye. We carefully adjusted index terms so that they were all less than 64 characters long.

2. The daily printoff of accession entries for the cataloguer to check was prepared in a relatively compact format. For ease of use, the layout should not be so compressed as to be difficult to read, nor so spread out that the reader must deal with (and eventually discard) large amounts of paper. We adjusted the layout of the check printout several times before settling on one which seemed to work reasonably well.

The collection of material was not processed in date order, so on any one day the cataloguer would deal with a number of types and years of material. The accession numbers served to segregate the catalogue entries by type and year. This breakdown was preserved for storing the catalogue entries. Our computer (5 years old at the time of the project) had only 2 disk drives with individual capacity for only 75K characters of storage. To keep each file of accession numbers small enough so that catastrophes of human or machine origin would have minimal effect, we arbitrarily imposed an artificial limit of approximately 2500 characters per file. This was quite close to the amount of data for a single type of material and year of origin which might be catalogued in one day. The positive benefit of files which were small, easily edited and "safe" against disaster was countered by the large number of such files, with backup copies, spread over 8 diskettes. We overcame this file management problem with a series of small BASIC programs which used local extensions to the language so that the disk directory could be read. Thus we could print inventories of the accession numbers used, with reports of missing numbers or duplicates. We could sort the files, which used names like S76 for slides in 1976, or S76B for the second in a series of such files, and even automatically consolidate all such files for a given type of material and year. A simple in-memory sorter was also developed to organize the final accession files so that each was in the proper order.

The accessions list, in final printed form, had 86 pages for negatives, 53 for photographs, 34 for slides, 3 for films and 1 for videotapes. To ensure that the printing could be smoothly picked up after a paper jam or other minor disaster, the accession printing program was based on the following principles:

A template for each record was constructed in memory. Index numbers were augmented with their corresponding terms (the index files were small enough to keep in memory for this operation).If sufficient space was left on the current page, the record was printed; otherwise, a new page was started, then the record was printed. This means no record crosses a page boundary, and considerably simplifies the task of restarting from a given page number or accession number. Pages were numbered within a given material type (or index),that is, we did not use global pagination in the catalogue, but each page is titled e.g. Negatives - 16.

The indices were built in a two stage process. First, a quite large but simple file containing records of the form

(index number) (delimiter) (accession number)

e.g.

G32~S82-21

implies that the accession record for the 21'st slide catalogued for 1982 refers to the 32'nd geographic index term. Through building this file by looking at films, negatives, photographs, slides and videotapes in that order, and within these types of material keeping to a date order, the file of index numbers is in the correct accession order and does not need resorting.

Display of the indices requires a program which can collect all the accession numbers which are appropriate to a given index term and print them in a presentable form. The program to do this was arranged to build up a part of a single index in memory by searching the simple file(s) described in the previous paragraph. The created index could then be output to a file for accumulation and editing into the final index.

6. JUSTIFICATION OF THE STRATEGY

Throughout the discussion above we have given our reasons for choosing particular approaches to the problem. We feel quite strongly that the type of project carried out is sufficiently common that others will arise for consideration by our company or others. Nevertheless, we feel certain that the particular details of each will be such that an attempt to prepare a generalized software package for cataloguing projects of this type is a waste of effort. Moreover, keeping the software for any appreciable length of time requires that it be fully documented. For the size of programs involved (the largest is about 3 pages long) the documentation takes as much effort as the programming and debugging. What we have preserved is a set of notes (from which this paper is drawn) pointing to the ideas which helped or hindered our progress. We also note that the most time-hungry part of the catalogue process was the generation of the partial index files i.e. the sections of the final indices which were later combined using word processing software. This is hardly unexpected, since sorting and searching are well-known to be tasks which are demanding of machine resources. However, the operation of the program to carry out the index building was almost totally automatic after appropriate selection of the type of index and range of terms to be found. The major issue for the operator was one of choice of alphabetic range of index terms to make best use of the available computer memory (the entire index segment was built up in memory, then dumped to disk).

Keeping all parts of the system "small" and manually verifiable helped us to overcome several obstacles, even an unusual hardware problem -- a broken resistor -- which arose near the end of the project. After a local shop was unable to locate the problem, we repaired the machine ourselves, though not without a great deal of trouble. However, secondary problems in the catalogue or indices arising from the fault were not difficult to overcome, since the maximum range of damage was limited by the small scope of both data files and programs, complemented by simple but effective manual controls.

We do not feel that the project could have been completed on time if we had used purely manual processes. Nor would we have delivered the catalogue when due if every program was prepared with complete documentation and checking. Use of a commercial database package would have involved some programming of output formats at the very least, and may have been prohibited by the multi-disk nature of the final database of catalogue records.

7. RESPONSIBILITIES

The classification scheme, cataloguing and document layout were carried out by Mary M. Nash. Carole Gaulin performed the data entry and disk control. Software, apart from the Anderson Technoproducts Word Processing package, was written by John C. Nash, who also maintained the hardware.

CAIS INFORMATION TECHNOLOGY
SPECIAL INTEREST GROUP (ITSIG)

Roddy Duchesne
Senior Network Officer
National Library of Canada
395 Wellington Street, Ottawa, Ontario. K1A 0N4

ABSTRACT

A resolution was passed at the May 25, 1983 CAIS Annual General Meeting to the effect that CAIS should form a special interest group or committee with a mandate relating to Information Technology. ITSIG is now (May 1984) in process of formation. The paper outlines the background, objectives, initial program and organizational structure of the Group. The paper seeks to provide a basis for discussion and agreement of final details; it also seeks to provide enough information for readers to decide whether they wish to participate and to suggest nominations for the ITSIG Executive and Electronic Working Group.

Le groupe d'intérêt spécial sur la technologie de l'information de l'ACSI (ITSIG).

Une résolution a été adoptée à la réunion générale annuelle de l'ACSI tenue le 25 mai 1983 à l'effet que l'ACST doit établir un groupe ou un comité d'intérêt spécial avec un mandat se rapportant à la technologie de l'information. L'ITSIG est maintenant (mai 1984) en train d'être formé. La publication souligne les antécédents, les objectifs, le programme initial et la structure organisationnelle du groupe. Elle cherche aussi à pourvoir assez d'information aux lecteurs afin qu'ils puissent décider s'ils veulent participer et suggérer des candidats pour l'exécutif de l'ITSIG et pour le groupe de travail sur l'électronique.

CAIS INFORMATION TECHNOLOGY
SPECIAL INTEREST GROUP (ITSIG)

1. INTRODUCTION

The following resolution was passed at the CAIS Annual General Meeting on May 25, 1983:

> "That CAIS should form a special interest group or committee with a mandate relating to Information Technology and should be more active in relation to Information Technology."

The CAIS National Executive subsequently agreed a set of guidelines for the formation of Special Interest Groups. An outline ITSIG proposal following these guidelines was then drawn up and approved by the CAIS National Executive. The present paper is a reformatted version of the document considered by the National Executive; its purpose is to outline ITSIG proposed objectives, initial program, and organizational structure to provide:

a) a basis for discussion so that details can be agreed and a more complete final version can be prepared for approval by the National Executive. Examples of additional details to be included in a final version are the names of the ITSIG Executive and Electronic Working Group, and a first year budget.

b) enough information on ITSIG for readers to decide whether they wish to participate and to suggest nominations for the ITSIG Executive and Electronic Working Group.

A working definition of the term "information technology" (IT) is given in Appendix A.

2. ITSIG OBJECTIVES

2.1 Primary objectives

a) Assist the interchange within CAIS of IT information and experience, and of views and concerns relating to IT issues.

b) Assist the interchange between CAIS and other national and international bodies (e.g., CIPS, ASIS) of IT information and experience, and of views and concerns relating to IT issues.

c) Gather IT-related information, experience, views and concerns and make these available as appropriate.

d) Advise the CAIS Board on IT issues.

e) Carry out assignments relating to IT at the direction of the CAIS National Executive.

2.2 Secondary objectives

a) Mount and maintain a program with national-level professional visibility.

b) Explore possibilities for cooperation with other professional bodies (e.g., ASIS, ASTED, CIPS, CLA) concerned with information technology.

c) Make use of new information technology to further work towards other ITSIG objectives, where this is feasible and cost-effective.

3. OUTLINE INITIAL ITSIG PROGRAM

The proposed first year program includes:

a) ITSIG formation.

b) Consideration of a feasibility study for mounting an on-line directory of existing and planned microcomputer, videodisc and videotex applications.

c) Subject to (b), participation in the implementation of an electronic directory accessible on a nationwide basis.

d) Organization of a CAIS Conference 1985 ITSIG session with a theme of (provisional wording) "implementing microcomputers in information centres". Subject to (b) and (c) above, the electronic directory would be described in one part of this session and would be available for access from terminals in an exhibit area. The session would also likely include:

  (i) A presentation generally reviewing microcomputer technology, and its applications in information centres.

  (ii) One or more presentations providing case studies of implementing mirocomputers in information centres.

(iii) One or more presentations on applications of other information technology such as electronic mail, videodisc, and videotex.

(iv) A paper on IT issues.

4. ORGANIZATIONAL STRUCTURE

4.1 General

ITSIG will comprize:

a) ITSIG Executive.

b) Electronic Work Group.

c) General membership.

These are reviewed in Sections 4.2 - 4.4 below.

4.2 ITSIG Executive

This will comprize the following positions:

a) Coordinator.

b) Secretary.

c) Membership Coordinator.

d) Program Coordinator.

e) Communications Coordinator.

f) Treasurer.

The holders of these positions will all be members of the Electronic Working Group and will all be available for ITSIG Executive meetings which will normally be monthly. In practice, this is likely to mean that Executive members will all live and work within easy reach of a common city. If practical projects (e.g., the electronic directory project) are initiated, the additional position of "Project Coordinator" will be added to the ITSIG Executive.

4.3 Electronic Working Group

This will comprize:

a) ITSIG Executive - outlined above.

b) Nationwide membership.

c) Chapter representatives.

d) Liaison persons nominated by other national and international bodies such as ASIS, ASTED, CIPS and CLA.

The criteria for initial membership of the nationwide group will be:

a) Expertise, experience and involvement in some aspect of information technology.

b) Willingness to use electronic mail on a regular, preferably at least daily, basis.

c) Paid up membership of both CAIS and CAIS ITSIG.

d) Invited by ITSIG Coordinator.

Chapter representatives will be nominated by Chapter Presidents. They will be persons who make regular daily use of electronic mail and who are Chapter members. Preferably they will also be members of their Chapter executives.

Nomination of liaison persons by other national and international bodies will follow communication between the CAIS President and the Presidents/Chief Executive Officers of the relevant bodies.

4.4 General ITSIG Membership

This will be open to any CAIS member who fills out an ITSIG application form and pays the ITSIG membership fee of $10. For this fee, the member will receive each year:

CAIS INFORMATION TECHNOLOGY SPECIAL INTEREST GROUP (ITSIG)

a) Two issues of the ITSIG hard copy directory which will list ITSIG members, their addresses and telephone numbers, electronic mail IDs, plus IT applications in which they are directly involved. The directory will be useful for locating and messaging other ITSIG members involved in relevant IT applications.

b) A copy of at least one draft ITSIG conference paper. In the first year this will be a paper on implementing microcomputers in information centres - with the opportunity to comment. This will eventually appear as a CAIS Conference 1985 paper but ITSIG members will see it much sooner. Where appropriate, their projects and papers may be referred to in the ITSIG paper. A draft-paper on IT issues may be treated in the same way.

c) Instructions concerning how to access and use the online ITSIG "Infotech" directory, and how to contribute entries to this directory.

d) The opportunity to submit papers for personal presentation within the CAIS Conference 1985 ITSIG session.

CAIS INFORMATION TECHNOLOGY SPECIAL INTEREST GROUP (ITSIG)

APPENDIX A

INFORMATION TECHNOLOGY

For working purposes, the following definition is employed:

Information Technology ('IT')

Refers to methods and systems for information creation, processing, storage, access, distribution, and communication. New methods are typically associated with microelectronics and related developments in communications and computers and their applications. These applications typically impact the office, home, and communications/manufacturing/industrial environments. A short list of illustrative examples includes micro-, mini-, mainframe-computers; telephone, micro-wave, cellular radio, cable, fibre optic and satellite communications; copy machines, word processors, facsimile transmission, electronic mail and other office automation/communication developments; television, radio and videotex; data base and value-added network services; robots and distributed process controls; videocassette, magnetic tapes and discs, videodiscs and optical data discs. While this illustrative list is primarily a list of technologies, it is noted that many significant information technology developments have important economic, legal and regulatory aspects. The dramatic decreases in the cost of computing power which have taken place over recent years provide an example of an important economic aspect.

The above broad definition of the term "information technology" aligns with recent Federal Government of Canada publications, e.g., *Culture and communications*. Montreal, Supply and Services Canada, 1983, pp.3-6.

INTEGRATION OF TELIDON WITH BASIS -- A QUERY-DRIVEN BIBLIOGRAPHIC UTILITY

Charles P. Cohen

INFOMART
164 Merton St
Toronto, Ont.

Current affiliation:

Wood Gundy
Royal Trust Tower, Box 274
Toronto-Dominion Centre
Toronto, Ont. M5K 1M7

ABSTRACT

It is difficult for untrained users to search large or highly-structured databases. The standard tools available have been complex query languages (e.g. BASIS) or structured database traversals (e.g. Telidon trees). We suggest and give examples of a search technique which enables naive users to answer their questions without knowing anything about the underlying structure of the database or its native query language. This technique has been applied successfully to the Canadian Record Catalog and the Secretary of State Terminology Databank.

L'intégration de Télidon et de Basis: un système bibliographique de type «Question-réponse» Revue.

Il est difficile pour des utilisateurs non-entraînés de consulter des bases de données importantes ou hautement structurées. Les outils standards disponibles ont été des langages de «demandes» complexes (ex. BASIS) ou des bases de données à structures «transversales» (ex. «l'arbre» Télidon). Nous suggérons et donnons des exemples de techniques de consultation qui permettent aux utilisateurs naïfs de répondre à leurs questions sans rien connaître sur l'infrestructure de la base de données ou de son langage de «demandes» originel. Cette technique a été appliquée avec succès au Catalogue Canadien des Procès-Verbaux (Canadian Record Catalog) et à la Base de données de terminologie du secrétariat d'état.

Integration of Telidon with BASIS

I. Rationale

As computers have become more powerful, they have also become easier to use (this is sometimes called "overhead" by old-line programmers). The user no longer needs to know how a computer works -- a day's training in BASIS, Inquire, Dialog, ... is sufficient to achieve minimal competence. But per must still:

- understand the structure of the database being searched
- be trained in (or otherwise learn) a weird language.

We were faced with a large online catalog of Canadian recordings, with a potentially large group of utterly unsophisticated (that is, untrained in searching) users. The state of the art was not good enough, and had to be changed.

II. Telidon

Telidon is two things:

- an excellent standard for the transmission of gra-phics and text (now called "NAPLPS")
- a method of structuring data into static "trees", where the nodes are "index pages" or "menus" with choices on them, and the leaves are "data pages" with information on them.

The static structuring makes it difficult to add new data, since index pages must be changed to lead to the new data pages. The tree structure also means that a wrong choice at any node (and there are lots of chances for wrong choices) means the user doesn't get per's desired information.

Most important, though, the fact the the tree is the only possible search mechanism makes many queries impossible. A tree of restaurants, in which

Location: 1) Central 2) East End 3) West End 4) Other

is an early choice, makes it impossible to find all pizzerias. And no way can a user ask for places which serve both shrimp and veal. In general, in a tree-structured database, a page has one index-term -- the concatenation of the choices which lead to it. This is like having a library with only one card catalog -- a subject catalog -- and no cross-references.

But it is so easy to use -- no training required . . .

III. BASIS

(Note -- the comments below, about BASIS, will apply mutatis mutandis to most other sophisticated retrieval packages. I am finding fault with the technology, not with a specific product within it.)

BASIS is a big, powerful, complex IR package -- term-switching, subfield indexing, free-text indexing with or without delimited keywords, sorting, relational-type traversals -- they're all included. The only problem is, to get a list of John Lovesin's LP's (this is in the Canadian Record Catalog), one must say:

```
>  FIND FS=A AND ART=LOVESIN
     11 documents found
>  SORT DR/D,TI/A
>  PRINT TI,DR,LBL,CAT (for title,  date of release,  label,
                        and catalog number)
```

This is fine for a trained searcher who works with BASIS every day, but not very nice for a record-store clerk. But the clerk needs this information, not the searcher. How to make the computer so smart that it can even understand a clerk -- that's the problem.

BASIS can save and re-execute search strategies (the saved strategy is called a "profile") and let the user specify arguments to them -- so a profile for the above search would look like:

```
>  FIND FS=A AND ART=[Enter artist's last name:]
>  SORT DR/D,TI/A
>  PRINT TI,DR,LBL,CAT
```

and it would be invoked when one said:

```
>  /* PERFORMER SEARCH   (or some similar sensible name).
```

Now, our problem is reduced -- we write and store a bunch of profiles, and just have to teach the clerk the names of twenty-two pre-packaged search/sort/display combinations, and what each one does. Still not simple enough.

III. Search Strategy Trees

Our twenty-two search strategies (Why twenty-two? List all the ways a clerk might be asked for an album, or a single, or a cut on an album -- by performer's last name, performer's full name, composer, . . .) do fall into a kind of schema:

What do you want -- 1)album 2)song 3)biography

What do you know -- 1)composer 2)performer 3)title

How much of it -- 1)exact 2)some words 3)parts of words

We could document our profiles for the clerk by drawing a tree of choices -- if per followed the branches of the tree, per would arrive at the name of the profile that would search for the right thing, using the right locators, and per would enter that profile name to do the search within BASIS.

A "tree of choices" is precisely the way Telidon retrieval works -- but in Telidon, it's the data that's organized as a tree, not the search strategies.

As it turns out, "tree" is not quite the right structure -- "network" or "graph" is required to handle things like looping back to the main choice menu after a display.

IV. Implementation

We did not supply all users with paper flowcharts of choices and profile names. Instead, we built a user-friendly, fully-prompting front end which was general enough to handle a wide variety of IR applications. It included:

- a network-type database of search-strategy menus -- questions to ask the user to figure out what per was looking for and how per wanted to find it
- the ability to prompt the user for arguments to profiles and automatically send the search requests to BASIS -- the user, having defined per's search strategy, finally sees:

  Please enter performer's last name: LOVESIN

  which is sent as an argument to the appropriate profile.
- display procedures for records selected by a search
- routines for following pointers embedded in the database records -- that is, a user could use keyword searches and Telidon trees in the same database
- graphic (NAPLPS format, naturally) displays for search menus and data, and stored graphics.

This has recently been extended with:

- testing/setting "system variables" -- e.g., number of records selected by previous search, user language, terminal type, etc.

The network database of questions which define searches and prompt for their arguments (one needs about 30 "question pages" for the Canadian Record Catalog) is, itself, a BASIS database, and can be edited and extended using packaged tools.

This software has been applied to the Secretary of State TERMIUM III databank, a 600,000-record multi-lingual terminology file.

V. Extensions

The technique suggested here is useful in situations in which:

- each record has many keys, and
- combinations of keys are often necessary, and
- common search strategies are predictable.

During development, we realized that our friendly structure was not restricted to BASIS databases -- the underlying information could be in a formatted DBMS, and the "profiles" could be replaced by packaged sequences of DBMS instructions -- things like:

```
LIST ALL TITLES VIA TITLE_ARTIST_SET FOR ARTIST WITH
     ARTIST.LAST_NAME EQ "LOVESIN"
```

(that's imaginary, but not by very much). It is likely that a large retrieval system for an unnameable client will be developed using search-strategy menus as the front end of a VAX DBMS-based database.

VI. Conclusion

We made a powerful retrieval system easier for untrained people to use. Our method works for different data structures and supporting software. It doesn't involve much run-time overhead, and it doesn't require any extra indexing of the underlying data. The framework of searches is stored in a database, and is easy to modify and extend. We think it's neat.

INFORMATICS AND INFORMATION DEPENDENCY

Victor M. Owen
Doctoral Candidate
Department of Sociology and Anthropology
Carleton University
Ottawa, Ontario, K1S 5B6

ABSTRACT

The impact of new technologies, especially what has been termed 'informatics', that is, the science of information technologies, remains relatively free from sociclogical enquiry. In our rush to cope with the economic and organizational changes resulting from these innovations there is a tendency to cast off the old for the new. The more immediate responses afforded by computers often makes our previous methods appear antiquated and cumbersome by comparison.

This paper addresses some social issues raised by the introduction of computerized banking, word processing and 'tele-shopping' made possible by the micro-chip and 'Telidon-type' formats. Some attention is given to similar applications of information technology in terms of its implications for existing copyright laws and publishing practices.

L'INFORMATIQUE ET LA DÉPENDANCE D'INFORMATION

RÉSUMÉ

Les répercussions des nouvelles techniques et, plus particulèrement, de l'informatique, c'est-à-dire, la science des techniques de l'information, ont été peu étudiées par la sociologie. Afin de s'adapter rapidement aux changements économique et organisationnels qui découlent de l'automatisation, l'on tend à remplacer le vieux matériel par du plus perfectionné. Aussi nos techniques anciennes semblent-elles désuètes à côté des ordinateurs, qui eux, livrent des résponses de façon instantannée.

Le présent travail praite des questions sociales qui se rattachent à l'implantation du système télé-banque, du traitement de texte et de l'achat par ordinateur, réalisable grâce à la technique des micro-plaquettes et du format de type 'Telidon'. Il sera également question des incidences d'autres techniques sur la loi sur les droits d'auteur et les procédés de publication.

INFORMATICS AND INFORMATION DEPENDENCY

INTRODUCTION

In light of the overworked clichés of our technological society viz. the 'dawn of the computer age', the 'information revolution', the 'third wave' and -- how could we forget with all the media hype -- 'Big Brother', it is difficult to posit more than a cautionary note on the literature which now inundates us. Moreover, any argument suggesting a second look at recent technological advances is apt to be labelled Luddite, anti-technology or unreasonably conservative by proponents of the new innovations.

What needs to be identified for the purpose of this paper is the specific rationale and framework used to view information technology, before we make any claims to insight. As information scientists we must first be concerned with the process of communication. Indeed, Wilhelm von Humboldt suggests that:

> *By the same process whereby he [man] spins language out of his own being, he ensnares himself in it; and each language draws a magic circle from which there is no escape save by stepping out of it into another.* (Cassirer 1946)

The problem we face now is the new language of 'technocratese' -- the language of an information society, or more specifically, the language of an emerging elite within that society. This elite is composed of the computer literate -- both the information high priests and the technical professions and industries which produce and maintain the computer/communications systems. Each sector of this group is compelled to devise new terms to fit new innovations. Acronyms, therefore, are not only the fanciful device of marketing personnel but the necessary product of a limited language. How we come to ingest these signs, however, is another issue.

A second phenomenon which we must identify to frame this polemic concerns the complexity of an information society; a complexity which is abetted by the concept of 'miniaturization'. (Bell 1980) As a result of this miniaturization process a whole new set of social relationships has emerged both in terms of the labour force within industries and our concept of *power* and *controls*. Having said this let us now turn to the issues presented by this new era of technology.

INFORMATICS AS DEFINITION

Informatics is foremost the study of information systems and how they work. Godfrey *et al* defines the term as 'information science' in both a theoretical and applied sense, that is, *"the collection, classification, storage, retrieval and dissemination of recorded knowledge"*. (Godfrey 1980) Indeed, 'information science' is a particular process and information scientists are engaged in the research and implementation of this process. What this definition fails to clarify is *what* is to be collected, *how* and *where* it is to be stored and for *whom* and for *what purpose* it is to be used. In short, I suggest we have concentrated most of our efforts on the technological questions related to the system and means of communications rather than on content and meaning. Informatics, moreover, is but another in a series of coined terms which attempt to convey meaning for the information technologies. For example 'telematics', borrowed from the French 'télématique', refers to the merging of telecommunications and computer systems. Still others choose to include information, information technology and communications. (Ryan 1981) Further confusion is added by what Oettinger calls 'compunications' to refer to the merging of telecommunications and teleprocessing, (Bell 1980) -- not unlike the European télématique.

In practical terms the struggle for definitions is not so much a question of semantics as it is an indicator of the confusion inherent in the evolution of these composite technologies. Such merging of technologies and services cannot be viewed as being simply the product of individual technologies anymore than the automobile is the product of the horse and carriage. Telidon, for example, creates a new set of services and expectations different from those using television, the computer, and telephone/cable service alone. These individual components are, indeed, 'familiar symbols' yet they call for a new set of responses in the Telidon context. (Tombaugh 1982) What becomes important for us is the way in which the technology is perceived, for ultimately this results in how we use the technology and how we define regulatory policies and legal controls.

The paradox which persists is that, for the most part, we continue to regard the new technology with 'old eyes'. Train tracks evolved in the path of wagon wheels, automobiles were originally called 'horseless carriages' and television began as 'radio with pictures'. The same holds true for informatics. Take, for example, the development of the Telidon protocol. Its original design was intended to provide a means of minimizing the quantity of data required to be transmitted between two points by conveying only the control instructions which would affect identical database files. (Bown 1978) Ironically, we now find that many of the commercial applications limit the technology to the alpha-numeric display of text or, in the other extreme, background pictures to enhance limited statistical information used in weather forecasts. In the former case the technology is underutilized, in the latter it is perhaps superfluous for most interests.

INFORMATION DEPENDENCY

What becomes apparent with these applications is that we often attempt to load our new system with old baggage even to the point of turning television sets into electronic books. Instead, we might make better use of the medium to convey the range of literature or its location. In this context we are not dealing with information but data which *"by themselves have no meaning"* as Sieghart suggests, and *"can only become information when they are combined together in some form which can increase the knowledge of the receiver"*. (Sieghart 1982)

Yet the capacity to store and retrieve vast quantities of information regardless of the quality remains central to most communication systems. This phenomenon is evident with the demand for multi-channel television. Among the more recent of informatics services are those presented by computerized banking. When the teller and customer become the components of the financial transaction, each compelled to follow set procedures in order to facilitate the process, there is an understood relationship in terms of each other's role. In this instance the relationship is one of human dependency with some understanding of the other's limitations. In contrast, the automated teller responds (provided the system is working and the customer is proficient with its operation) without bias and, presumably, without the normal delays required by a manual searching of accounts. This, then, is the second influence produced by informatics - an expectation of immediacy. A similar situation exists in in the case of tele-shopping, which replicates the catalogue order procedure while eliminating the need to interact with a salesperson, or for that matter, the need to travel to a shopping centre. There are both advantages and disadvantages to such procedures. What should be recognized, however, is that wide-scale adoption of this system will have far reaching social implications requiring the development of new alternatives for social interaction.

LABOUR AND EMPLOYMENT

Yet another arena for informatics technology involves those people directly working with the processing of information -- the computer operators, programmers and information librarians. In recent years we have witnessed the incredible changes in personnel requirements as a direct result of office automation, computer storage of information and cybernetics. There continues as a result a need for redefinition of the roles in this new environment, particularly for the office of the future (Licker 1982) and the information marketplace. (Thompson 1981) Although we would all like to believe that the deluge of information now created will provide the need for more employment, informatics technologies are not labour intensive. Moreover, computer technologies often facilitate the process of redesign and modification to existing procedures which could eventually spell the demise of many programming functions.

At the same time, there is a growing awareness that the threat of centralized and controlling mainframe computers has been eroded by the development of the mini and micro computer systems; the latter development giving rise to a new cottage industry and workplace. In turn, if this structure continues there will be a greater emphasis placed on family solidarity, with less direct social interaction produced by one's employment. Contrary to living in McLuhan's 'global village' we may find ourselves, as Munford once suggested, at the point of full circle in which electronic communication will differ from direct intercourse *"only to the extent that immediate physical contact will be impossible"*. (Mumford 1970)

COPYRIGHT PRACTICE AND POLICY

With the advent of photocopying machines, electronic recorders for audio and video information and the sheer availability of publications and media broadcasts, the issue of 'ownership' of information is an important concern, particularly for the producers of the information. For the most part, the use of information in print format or as media is covered by existing copyright laws in Canada. At the same time, it is becoming increasingly more difficult to police the unauthorized use of information. The F.C.C. in the United States recently ruled to permit 'off-air' recording of broadcasts since it would be impossible to control the already widespread practice.

It has been argued further that since:

> *The principle characteristic of 'property' is that, by stealing it, you can deprive the lawful owner of its use and enjoyment...there can be no property in information.* (Sieghart 1982)

What needs to be qualified, however, is the notion of property, that is, what should be considered personal property and what must be regarded as public by right of its content and use. Moreover, informatics is not limited by language or political boundaries. For example, Arthur C. Clarke's recent book 2010: Oddyssey Two was written on an Archives III micro computer, using 'Wordstar' software, then mailed from Colombo, Sri Lanka, to New York on a five-inch diskette and subsequently revised by satellite transmission through the Padukka Earth Station and the Indian Ocean Intelsat V. (Clarke 1983)

CONCLUSION

There comes with the new informatics a promise of better and faster methods for handling information -- assuming, of course, that this deluge can be digested at the same rate. However, it cannot. Not only is society by definition a complex combination of many people, cultures and traditions but it includes the artifacts, technologies and history of its people. This 'cultural baggage' shapes the use we make of information and through our knowledge and experience we give meaning to otherwise insignificant data.

REFERENCES

BELL, Daniel. The Winding Passage. Cambridge, Mass., ABT Books, 1980, pp. 35-39.

BOWN, Herbert G. A General Description of Telidon: A Canadian Proposal for Videotex Systems. Ottawa, Ont., Department of Communications Technical Note No. 697-E, December, 1978.

CASSIRER, Ernst. Language and Myth. trans., Susanne K. Langer, New York, Dover Publications Inc., 1946, p. 8.

CLARKE, Arthur C. 2010: Odyssey Two. New York, Ballantine, 1982, p. 291.

GODFREY, David, ed. et al. Gutenberg Two, Toronto, Press Porcépic Ltd., 1980.

LICKER, Paul. "Information Careers in the Office of the Future" in *Proceedings of the Tenth Annual Canadian Conference on Information Science*. Ottawa, Ont., May 1982, pp. 5-13.

MUMFORD, Lewis. *The Myth of the Machine, Vol. II: The Pentagon of Power*. New York, Harcourt Brace Jovanovich, 1970, p. 295.

RYAN, Michael. "The Social Impact of Telematics", unpublished paper presented at the Telematics and Higher Education Colloquium, Halifax, N.S., January 1981.

SIEGHART, Paul, ed. *Microchips With Everything: the consequences of information technology*. London, Comedia Publishing Group, 1982, pp. 11-12.

THOMPSON, Gordon B. "Ethereal Goods -- the Economic Atom of the Information Society", a paper presented to the E.C.C. Conference on the Information Society, Dublin, Ireland, November 1981.

TOMBAUGH, J. W. *et al*. "Evaluation of Graphics on Videotex by Inexperienced Users", in *Proceedings Graphic Interface '82*. Toronto, May 1982.

PHYSICIAN'S COMMUNICATION SYSTEMS: COPING WITH INFORMATION OVERLOAD

Joanne L. Warren
Communication Consulting
Convergence
507 Oxford Street East, Suite Two,
London, Ontario N5Y 3H8

ABSTRACT

This paper reviews research on physicians' information overload, integrating it with an overview of the different communication systems within which physicians acquire information: libraries, mass media, interpersonal networks and computer systems, examining their advantages and problems. It discusses some of the solutions to information problems in three general areas: changes in the professional media, new forms of information packaging through the help of information specialists, and innovations in microcomputers.

Des systèmes de communication pour médecins: Venir à bout des surcharges d'information.

Cette communication se veut une révision des recherches sur la surcharge d'information chez les médecins, en plus d'être une revue des différents systèmes de communication par lesquels les médecins acquièrent l'information: bibliothèques, media de masses, réseaux interpersonnels et systèmes d'ordinateurs, examinant leurs différents avantages et problèmes. Certaines solutions sont discutées pour les problèmes d'information dans trois secteurs généraux: des changements dans les médias professionnels, des formes nouvelles «d'emballage» de l'information avec l'aide de spécialistes de l'information et des innovations dans les micro-ordinateurs.

INTRODUCTION

Information overload is recognized as a generalized problem, but it is pronounced among scientists, physicians and others who must deal with a continuously growing and changing volume of facts. It is a very real phenomenon which is increasingly having an effect on the medical profession, specifically on programs and initiatives in continuing education. It is further complicated by a process of contamination, identified by Goldfinger (1982), in which competing suppliers of information attempt to provide the same educational message in different ways.

This article reviews the current literature on medical communication systems and information overload, identifying how physicians use the various systems, and how they cope with the stresses induced by overload. It suggests further approaches to solving the problem.

While there is no debate over the existence of information overload, there is not yet a generally accepted definition of the problem. It has variously been defined as: too much information in a qualitative and/or quantitative sense; a problem in the quality, complexity or novelty of information; or, more generally, as sensory overload. (O'Reilly 1980, Milgram 1970, Miller 1960, Lipowski 1974)

Albeit not clearly defined, the problem is rooted in the sheer volume of information available. In 1957, there were 3,879 serial titles and 218,734 articles published in medicine. At that time, it was estimated that, in order to keep abreast of the U.S. medical literature alone, a physician would have to read over 148 journals a day, seven days a week. (Harris 1966) Today, the problem is even more pronounced, with approximately 4,200 medical serial titles in print and over 30,000 biomedical and health-related journals. The problem is overwhelming within specialties; for example, the Journal of Immunology in 1982 published 985 papers in 5,703 large format pages. (J of Im Indexes 1982)

The overload that exists in printed communications is compounded by the continual provision of medical information to physicians in a variety of other media--radio, television, and direct exchange between doctors.

PHYSICIANS' USE OF COMMUNICATIONS SYSTEMS

Medical Libraries

The health sciences library is the centre of management systems and services for biomedical, state-of-the-art information. It is a far cry from the dusty storehouses of the 60's. In the past 20 years, there has been a quiet revolution in medical libraries--through the development of data-management systems involving knowledgeable staff, increased collections, and sophisticated data-retrieval systems such as MEDLARS.

Furthermore, the world's knowledge base is shifting inevitably from paper to electronic media at nearly a four percent compound rate annually (Matheson 1982), and biomedical knowledge is no exception. Therefore, much important, dynamically changing information is likely to be available only through data-processing systems. The academic medical library, with its resources for searching multiple biomedical data bases, can now provide invaluable resources that are available in libraries as a result of these innovations.

In 1972, a study of the patterns of self-education of 390 physicians, interns, and residents in Toronto revealed that less than one-half of them knew how to use Index Medicus, and one-quarter had never heard of Excerpta Medica. (Woodsworth 1972) Another survey reported in 1972 that, when 622 physicians were asked about their use of medical information sources, only 9.1 percent reported turning to the library for help. (Cameron 1973)

JOURNALS

Physicians describe print as the most important formal source of medical information (Bryans 1981, Curry 1981, Vollan 1955) despite evidence that many devote little time to reading medical literature. Kotre (1971), for example, reported that Michigan physicians spend an average of three hours or five percent of their working week reading the medical literature; Van Dellen (1961) found that the "average physician" reads for approximately 45 minutes a day. Nevertheless, survey results indicate that journals remain the most important medium of communication between physicians. In one study, 79 percent of physicians contacted in Ontario rated journals as important or very important (Cameron 1973); a recent survey of Ontario physicians (Bryans 1981) concluded that 99 percent had access to journals and read them 2.5 to 3 hours a week.

The breadth of physicians' reading has also been assessed. A study of physicians in a Health Maintenance Organization in the upper midwestern United States (Christensen 1979) showed that each routinely read a median of four journals, and that 30 percent read five or more. Collins and Coffey (1981) found that 75 percent of physicians surveyed in Alabama read one to five journals a week (sic), and 21 percent read six or more. Kotre (1971) reported that, in a Michigan survey, physicians typically subscribed to three or four medical journals.

In evaluating the depth of reading by physicians, Kotre (1971) found that 53 percent of the doctors he studied in Michigan read selected articles completely, 33 percent skimmed through journals, six percent read only abstracts or summaries, and none read journals cover-to-cover. In a study of physicians in six American states, Caplow (1952) found that eight percent read periodicals cover-to-cover, six percent scanned them systematically and read advertisements in detail, 46 percent skimmed in an unplanned manner, and seven percent reported that they did not read journals at all. Clearly, the use of print for learning is variable.

Mass Media

Mass media are open systems of communication that are directed to heterogeneous audiences, in contrast to the specialized focus of trade or professional journals. Between these two extremes are the special interest television and cablecast programs that are directed to physicians, but also attract an interested, nonprofessional audience.

There has been continual controversy about the desirability of transmitting technical, medical information through the mass media; indeed, many physicians generally fear and disdain the lay media (Altman 1975). The reasons for this are that mass media presentations may prompt patients to seek treatment that the physician is not prepared to offer (Brander 1975), these presentations may also portray physicians improperly as risk-takers (Gerbner 1981), and they are frequently incomplete, superficial, misleading, sensational, or lacking in critical judgment (O'Keefe 1970). In addition, mass media can serve as "a professionally uncensored route through which scientists and clinicians wishing to bypass peer review may bring their message directly to the public." (Brady 1975) Nevertheless, many mass media reports are well-researched and based on facts derived from medical journals and respected experts (Crawford 1982).

However controversial, the mass media do provide a quick and continuing form of medical education to both physicians and the general public. Interviews with 150 general practitioners and specialists (Gutman 1977) revealed that 93 percent were frequently exposed to some mass media source. Seventy-two percent watched television news; 48 percent frequently read health-related articles in newspapers; approximately 25 percent learned about health-related shows with other physicians. In 1967, Shaw and Van Nevel (1967) surveyed full-time members of faculty at the University of Wisconsin Medical School and found that 60 percent of the respondents reported learning new information in their own specialties from the mass media; 92 percent of the information came from print media such as newspapers and magazines; only two percent came from radio or television. The authors speculated that "it may be that medical science stories in magazines and newspapers serve(d) as a kind of 'index' to new developments for the busy specialist, alerting him to areas where he might direct his more detailed journal reading."

O'Keefe (1970) extended Shaw and Van Nevel's work, increasing the number of variables and using a broader sample of physicians in medical schools in North Carolina. He reported that 89 percent learned about medical developments from the mass media--some as frequently as once a week. While 73 percent of the respondents believed information from all the mass media to be "fairly reliable" or "very reliable," they still rated newspapers and magazines to be the most useful sources of information provided by the mass media. In a survey reported in 1977 (Gutman), 53 percent of the physicians canvassed in Michigan rated mass media sources as "useful," and 76 percent said that they did not believe it was unprofessional to use the mass media as a source of medical information. Indeed, 52 percent of the respondents thought that the mass media were "credible source(s) of information," and 21 percent reported that items from the mass media could influence their attitudes about health. Finally, almost one-quarter of the respondents wished that the mass media would present more health information that was applicable to their practices. Thus, research has shown that the mass media can be an important source of new medical information for the physician, but the nature and the extent of the impact of the media on the practice of medicine remain unclear.

Interpersonal Communication Systems

One of the most powerful medical communication systems--the interpersonal systems that exist alongside such formal channels as journals and conferences (Waksman 1980, Menzel 1955, Coleman 1966, Katz 1963, Brandjes 1976, Ryan 1977, Weinberg 1981)--have been analysed by researchers in the social sciences. The term "invisible colleges" has been applied to collegial and friendship networks that function over great distances (Crane 1972, Lancaster 1978). Members of the networks communicate via telephone, correspondence and professional meetings, and exchange preprints, reprints and drafts of proposals. In doing so, they function as quick efficient providers of knowledge, and as opinion leaders. Regardless of the distances involved, opinion leaders typically function as interpersonal communication systems linking formal information systems with the users of information. Sociological studies indicate that the opinion leader has high status because of his contributions to the medical community and his access to information (Coleman 1966). Opinion leaders (Rogers 1971) also tend to have more education, are more cosmopolitan, and have greater contact with "change agents," such as pharmaceutical representatives, who supply new information. This simple, two-step flow of information becomes multi-step when a variable number of relays exist in the communication line from source to audience. In studying this process, Menzel and Katz (1955) found that some individuals received messages directly from media sources, and others were several times removed--the specific pattern being dependent upon interpersonal relationships, availability of the media, the type of message, and its salience.

Influenced by the multi-step model, Rogers and Shoemaker (1971) have described characteristics that distinguish interpersonal communication from communication by mass media. Although both are sources of information, mass media primarily provide knowledge, while interpersonal contacts are more powerful in influencing attitudes.

Although research has revealed that interpersonal communication networks satisfy multiple information needs for the physician and have the advantage of being quick, as a system of accurate medical information, they have severe limitations as Green (1978) has pointed out, " . . . communication with one's colleagues may transmit accurate new knowledge but in the process may also involve transmission of a great deal of false information. Indeed, the latter would seem to be substantially more frequently the case than the former."

The New Media: Telecommunications and Computers

The term telecommunications, in this context, means the exchange of information between health-care workers using electronic media over a distance (House 1981). This system of transferring information has advanced rapidly from the early experimental use of audio-only shortwave channels (Woolsey 1958), through telephone (House et al. 1981) and radio-conferencing (Woolsey 1958) to single-frame and continuous, live, interactive televisions (Fisch 1972) linked by cable and/or satellite to many centres. There have been several studies of the effectiveness of these new types of communication.

In a 1958 study (Woolsey), 95 percent of the U.S. physicians surveyed rated the use of two-way, closed-channel radio, an early form of teleconferencing as good to excellent for continuing medical education. The researchers attribute part of the program's success to the state of anonymity given to the physicians who wished to ask questions of the moderator, but the absence of visual material reduced the participants' concentration. Since then, telephone conferences linking physicians in Newfoundland hospitals have been supplemented by precirculated visual material (House et al. 1981). This audio-visual format is interesting, speedy, economical, and convenient. Live, interactive, televised transmissions have also been used successfully to link hospitals in southern Ontario to remote locations (House 1981, Corey 1979, Carey and Russell, Carey et al. 1979, Conrath 1977, Dunn 1980, House 1977, Roberts 1977, Sophianopoulos, 1976).

In Hamilton, Ontario, a two-year experiment in interactive television included a presentation, a discussion between guest and host, plus a phone-in, question period. Eighty percent of the surveyed members of The Hamilton Academy of Medicine viewed these broadcasts at least once (Smith 1970), and 23 percent watched regularly during the first season. During the second season, 43 percent watched. Eighty-six percent rated the experience as good to excellent.

Computers have made it easier to organize and gain access to data for many reasons. Computers can perform simultaneous bibliographic searches quickly and cheaply (Graham 1981, Ledley 1966, Haas 1982), and information specialists can use them to provide the medical specialist with timely, comprehensive, tailormade information (Stableford 1982, Hillman 1969). Denfield suggests that, with the computer to assist him, the physician would have more time to devote to social and human values in his practice (Denfield 1977). Albeit a powerful tool, physicians have been reluctant to use the computer; they apparently fear a loss of professional status and lack faith in the technology (Schwartz 1970). However, those who have used computers, have evaluated them positively (Morrell 1977).

THE RESPONSE TO INFORMATION OVERLOAD

Over the years, physicians and scientists have developed a variety of defense mechanisms against the effects of information overload, ranging from filtering mechanisms to complete withdrawal (Brody 1975). A good example of a filtering mechanism is the referee system which selects information to be published, consequently reduced information flow and, presumably, enhances the quality of what ultimately appears in print (Brody 1975). Reviews, however, slow the process of information transfer and, it is well known, often have delayed the publication of key discoveries. Paradoxically, the journals, which now contribute to information overload, were developed in the 19th century in an attempt to overcome it. As Lancaster and Smith (1979) have noted:

> . . . the original purpose of journals was not to publish new scientific papers so much as to monitor and digest the learned publications and letters that were too much for one individual to cope with in daily reading and correspondence. In turn, when the number of journals grew too large for one person to monitor, the abstract journal emerged in professional societies.

Far from reducing the problem, the emergence of the abstract journal has actually contributed to it. For example, Chemical Abstracts published its first million abstracts over a period of 32 years; by the 1970's, it published a million abstracts in little more than three years (Lancaster 1961). In many cases, according to Waksman, "the real articles are now the abstracts." Recently, "letters journals" have appeared in an effort to speed the dissemination of early research results (Lancaster 1979).

Another attempt to deal with the problem has been the accelerated development of interpersonal networks of communication; in fact, the increasing mass of published material has led to a shift toward small, elite meetings and personal contacts (Waksman 1980). Physicians have often ranked highly either colleagues or pharmaceutical representatives as the original sources of new drug information (Coleman 1966, Stross 1979, Ferber 1958, Caplow 1954, Dresden 1959), and physicians rank pharmaceutical representatives first and other physicians second as sources that influence attitudes towards the prescribing of drugs (Coleman 1966, Ferber 1958, Gaffin 1956). The high rating of pharmaceutical representatives is surprising and may be because many physicians believe that pharmaceutical representatives are reliable sources of information inasmuch as they typically have some scientific background and knowledge of physiology, pharmacology, and the current literature (Dresden 1959).

Discussion

Today's physician not only feels the effects of information overload, but also stands at the juncture of two kinds of communication systems; the old one--a complex of print-based journals, the interpersonal systems, and the mass media--and a new complex of personal computers and data-base systems that potentially increase access to more journals and other sources of information. However, these new resources, like all previous innovations in disseminating scientific information, may increase the sense of information overload; but, generally, one can be optimistic because partial answers have been suggested in the literature. These include: increased use of the professional and mass media, the medical library, the use of local information specialists, and the computer.

1. The Media. An evolving fusion of the functions of the different media channels will possibly provide an answer to both underutilization and overload of information. For example, the problem of underutilization might be reduced by the addition of the general information from the mass media to the professional medical media. In fact, a recent editorial, commenting on the relatively small amount of time physicians spend reading professional journals, suggested that reading time could be increased by including nonmedical articles, thus creating "crossover" readership (Wood 1982). In this way, the medical journals might compete more successfully with the mass media for the physician's time, while the mass media might serve as an index to refer the physician to more detailed sources. Certainly, the physicians' positive reactions in the television experiment in Hamilton (Smith 1970) suggests a potentially successful fusion of medical sources and mass media.

2. New Forms of Information Packaging. Information overload creates the necessity for innovative ways of "packaging" information because, as has been pointed out, "Information needs have not been adequately identified or met by the traditional institutionalized librarian" and that information specialists must "reach out" and become "visible" to professionals in need. Information packaging means providing information geared to the specific needs of varied professionals; it is an active, rather than a passive approach, one that seeks to make information acquisition more conveniently available than it has been. This approach may involve creating roles outside of library and medical institutions that interface with them and with the practicing physician.

An example of this kind of involvement was demonstrated in research comparing two groups of cell biologists, one of which--the experimental group--was provided with information specialist service. At the end of ten months, there was a considerable difference, with the experimental group spending less time reading the literature, skimming articles, or talking with colleagues, but more time reading in depth, and reading more outside of their core area. The service also saved them time and identified relevant material that would not have been found otherwise (Neway 1982). Similarly, research at the Cook County Hospital in Chicago in which an information specialist made possible greater exposure of clinicians to current medical articles, patient care was improved by speed and access to the literature, and physicians learned to work independently in the library (Roach 1975).

The reference service of the Canadian College of Family Physicians, in the Sciences Library of The University of Western Ontario, is another example of a relatively recent attempt to package information. This library service is a convenient, computerized, easily accessible (by a simple phone call) system to supply up-to-date literature reviews, bibliographies, abstracts, and photocopies of new articles. However, only three percent of the 6,500 members of the Canadian College of Family

Physicians use it. Even so, the service is currently overloaded and can barely meet the demand. Thus, we are led to suggest a second kind of custom "packaging" that can be initiated by the physicians and small groups locally, by engaging a competent, free-lance librarian or information specialist to monitor the most sophisticated general and relatively specialized media, to flag articles of interest, and to provide regular, up-to-date bibliographies, reprints and selected condensations. Furthermore, the service could be "future-oriented," providing an index of future professional events, lectures, and meetings in specified topics.

3. The New Microcomputer Systems. Reports of brisk sales of microcomputers to physicians indicate that, potentially, they are increasing their ability to search general and special data-bases to secure up-to-date information rapidly. These systems are "user-friendly" and easy to operate, thereby enabling the practitioner to adapt sophisticated retrieval systems to his own needs--providing his own key words (e.g. diseases and medications) to call back notes or additional references instantly. Furthermore, these small systems can be linked to very large data-bases, such as those in health sciences and national libraries, for rapid access to extensive information. Recent findings of a national study of practising and academic physicians indicated that 42 percent believed a computerized data-base would result in a significant improvement in their access to medical information, compared with six percent who believed there would be no improvement (Singer 1983).

Eugene Garfield of the Institute for Scientific Information has described the new two-fold data retrieval system available now for such microcomputers as Apple (Garfield 1983). The system, called Sci-Mate, using automatic telephone modems, makes it possible for the busy practitioner automatically to almost create his own constantly up-to-date data-base by logging into giant data-bases such as Science Citation Index and Medlars. He could then key the material to his own research, or even to specific patient-care management.

This has some parallels with Lawrence Weed's "Problem-Knowledge Coupler," a computer program that, based on facts obtained from the unique patient, mobilizes information from the literature on a given problem, and then presents it to the physician and patient for further "coupling" with the patient's history and symptoms (Weed 1980). Weed has argued that such a system is necessary for optimal diagnosis because " . . . a single human mind cannot grapple successfully with all the complexities involved in defining and solving medical problems in the total context of an individual patient's life any more than single-celled organisms can do what a sophisticated multicellular organism can do."

Still another development extends and modernizes the influential if criticized interpersonal networks, and sharpens the advice and information provided by providing a printed record of it. The recently developed electronic conferencing techniques, such as Participate, available on a microcomputer data-base network called The Source, make possible a new kind of continuing medical education that occupies a niche between the concept of interpersonal communication networks and the new electronic journals--"a network of computers being used by people with common interests to exchange, evaluate and store information (Franklin 1982), mixing comments with actual journal papers, articles, and news items. A parallel example has been presented in a report by the Association of American Medical Colleges, in which the physician uses a "filtered" system--his office computer to contact a librarian for information, either urgent and specific to a case, or for general research, requesting simultaneous transmission to his colleagues with whom he may then consult in short order (Matheson and Cooper 1982).

Perhaps the ideal long-range solution for dealing with information overload would be the one proposed by Matheson and Cooper (Matheson 1982). This would involve the establishment of complex, highly-organized health sciences centres on university campuses. These centres, with the health sciences libraries at the core, would utilize the full range of technological and personnel resources at every level. They would be linked into a comprehensive medical resources network by a national agency, working in conjunction with other public and private organizations. The implementation of such a broad multi-levelled concept would surely go a long way toward dealing with the problem of burgeoning information. However, the ideas we have suggested here could be initiated at the individual, local and institutional levels rather quickly. The solutions we propose are different--ranging from new ways of using media, to the employment of information specialists and computers--but they share with the more comprehensive plan of Matheson and Cooper, the objective of helping the physician to cope with information overload by giving him a much greater sense of control and the ability to manage information in a selective, individual way.

REFERENCES

Altman LK. The mutual obligations of the medical profession and the press. RI Med J 1975;58:306-308,319.

Bander MS. Sounding board: the physician and the press. N Engl J Med 1975;291:402-403.

Brandjes JF, Kasowski MA, Davis MW. Information systems Part VI: what the computer can do for the primary care physician. Can Med Assoc J 1976;114:170-171.

Brody EB. The right to know. J Nerve Ment Dis 1975;161:73-81.

Bryans AM, Davis D, Delmore T, Dunn E, Heron A, Krauser J, Scott DJ, Williams JI. Continuing medical education of physicians in Ontario: practices, needs, and problems. Toronto: Faculty of Medicine, 1981.

7

Cameron WJ, et al. Medical information network for Ontario: determination of a need. London, Ontario: School of Library and Information Science, University of Western Ontario, 1973.

Caplow T. Market attitudes: a research report from the medical field. Harvard Business Review 1952;30:105-112.

Carey LS. A telemedicine experiment in Canada using Hermes. Proceedings of Telehealth '79. Winnipeg, Manitoba.

Carey LS, Russell ES. A telemedicine experiment in Canada using a satellite, Hermes: a telecommunications experiment between a remote nursing station (Kashechewan), a base hospital (Moose Factory General) and a health science centre (University of Western Ontario). London, Ontario: University Hospital, U.W.O.

Christensen DB, Wertheimer AI. Sources of information and influence on new drug prescribing among physicians in an HMO. Soc Sci Med 1979;13A:313-322.

Coleman JS, Menzel H, Katz E. Medical Innovation: a diffusion study. New York: Bobbs-Merill, 1966.

Collins GE, Coffey FL. A survey of drug information: needs of physicians in the state of Alabama. Ala J Med Sci 1978;1:91-97.

Conrath DW, Dunn EV, Loor WG, Tranquada B. A clinical evolution of four alternative telemedicine systems. Beh Sci 1977;22:12-21.

Crane D. Invisible colleges: diffusion of knowledge in scientific communities. Chicago: University of Chicago Press, 1972.

Crawford C. Broadcasting science and medical news: the responsibilities and obligations. Info 1982;57:9-14.

Curry L, Putnam RW. Continuing medical education in Maritime Canada: the methods physicians use, would prefer, and find most effective. Can Med Assoc J 1981;124:563-566.

Denfield KF. Library service to physicians in Washington State. Bull Med Lib Assoc 1977;65:381-382.

Dresden MK. An unpublished paper for Wharton School at the University of Pennsylvania, 1959.

Dunn EV, Conrath DW, Action H, Higgins C, Math M, Bain H. Telemedicine links patients in Sioux Loookout with doctors in Toronto. Can Med Assoc J 1980;122:484-487.

Ferber R, Wales HG. The effectiveness of pharmaceutical promotion. Urbana, Illinois: Bureau of Economic and Business Research, University of Illinois, 1958.

Fisch A, Dwyer TF. Interactive television in the continuing education of foreign-trained psychiatrists. J Med Educ 1972; 47:912-914.

Franklin J. Primary information on-line in biomedicine: an appraisal. Scholarly Pub July 1982.

Gaffin B, et al. The Fond du Lac Study. Chicago, Illinois: American Medical Association, 1956.

Garfield E. The Sci-Mate Universal Online Searcher. Curr Contents 1983;14:5-15.

Gerbner G, Gross L, Morgan M, Signorielli N. Health and Medicine on television. N Engl J Med 1981;305:901-904.

Goldfinger SE. Continuing medical education: the case for contamination. N Engl J Med 1982;306:540-541.

Graham DL, King C, Whitney PJ. Simultaneous remote search on-line bibliographic library services for practicing physicians. JAMA 1981;246 1115-1116.

Greene NM. Gossip and the acquisition of knowledge (editorial). Anesth Analg 1978;57:519-520.

Gutman J. Physicians' exposure to health topics through mass media: an avenue for improving the dietician's image. J Am Diet Assoc 1977;71:505-509.

Haas WJ. Computing in documentation and scholarly research. Science 1982:215.

Harris JJ. Survey of medical communications sources available for continuing education. J Med Educ 1966;41:737-755.

Hillman DJ. Biomedical information systems, in R. Cheshier Information in the health sciences. Cleveland, Ohio: Med Lib Assn, 1969.

House AM. Telecommunications in health and education. Can Med Assoc J 1981;124:667-668.

House AM, Roberts JM. Telemedicine in Canada. Can Med Assoc J 1977;117:386-388.

House AM, Roberts JM, Canning EM. Telemedian provides new dimensions in CME in Newfoundland and Labrador. Can Med Assoc J 1981;124:801-802.

Journal of Immunology Indexes, 1982.

Katz E, Levin ML, Hamilton H. Traditions of research on the diffusion of innovation. Am Sociol Rev 1963;28:237-252.

Kotre JN, Mann FC, Morris WC, Vanselow NA. The Michigan physician's use and evaluation of his medical journal. Mich Med 1971;70:11-16.

Lancaster FW. Toward paperless information systems. New York: Academic Press, 1978.

Lancaster FW, Smith RC. Science, scholarship and the communication of knowledge. Library Trends 1979;27:367-388.

Ledley RS. Computer aids to medical diagnosis. JAMA 1966; 796:933.

Lipowski ZJ. Sensory overloads, information overloads, and behaviour. Psychother Psychosom 1974;23:264-271.

Matheson N, Cooper J. Academic information in the academic health sciences center. J Med Ed 1982;57:1-93.

Matheson N. Roles for the library in information management. In Matheson N, Cooper J. Academic information in the academic health sciences center. J Med Ed 1982. 57, Appendix B: 77-85.

Menzel H, Katz E. Social relations and innovation in the medical profession: the epidemiology of a new drug. Pub Op Quart 1955;19:337.

Milgram S. The experience of living in cities. Science 1970; 167:1461-1468.

Miller JG. Information input overload and psychopathology. Am J Psychiatry 1960;116:695-704.

Morrell J, Podlone M, Cohen SN. Receptivity of physicians in a teaching hospital to drug interaction monitoring and reporting system. Med Care 1977:68-78.

Neway JM. The role of the information specialist in academic research. Online Rev 1982;6,6:527-535.

O'Keefe MT. The mass media as sources of medical information for doctors. Journalism Quarterly 1970;47:95-100.

O'Reilly CA. Individuals and information overload in organizations: is more necessarily better? Academy of Management Journal 1980;23:684-696.

Roach AA, Addington WW. The effects of an information specialist on patient care and medical education. J Med Ed 1975;50:176-180.

Roberts, JM, House AM, McNamera WC. Telemedicine at Memorial University of Newfoundland in Med Info '77 eds. Shires/Wolfes, North Holland Publishing Company:869-973.

Rogers EM, Shoemaker FF. Communication of innovations: a cross-cultural approach. Glencoe, Ill: Free Press, 1971.

Ryan JF, Murray JA. Diffusion of a pharmaceutical innovation in Ireland. Eur J Marketing 1977;11:3-12.

Schwartz WB. Medicine and the computer: the promise and problems of change. N Engl J Med 1970;283:1257-1264.

Shaw D., VanNevel P. The informative value of medical science news. Journalism Quart 1967;44:548.

Singer J, Sacks HS, Lucente F, Chalmers TC. Physician attitudes toward applications of computer data base systems. JAMA 1983; 249:1610-1614.

Smith SL, Barrows HS, Keast R. Physician and public interest in medical television broadcasts: a report on two years' experience. Can Med Assoc J 1970;140:1101-1103.

Sophianopoulos MM. Medicine in the North. Telesis 1976;4:258-262.

Stableford BA. Using computers to access pharmaceutical literature. Can Pharm J 1982:44-47.

Stross JK, Harlan WR. The dissemination of new medical information. JAMA 1979;241:2622-2624.

VanDellen TR. The functions of state medical journal. Address, State Medical Journal Conference, Chicago, Illinois, 1961.

Vollan DD. Scope and extent of postgraduate medical education in the United States. JAMA 1955;157:307-308.

Waksman BH. Information overload in immunology: possible solutions to the problem of excessive publication. J Immunol 1980;124:1009-1015.

Weed L. Problem solving in medicine. Can we teach it? (unpublished paper) July 10, 1980.

Weinberg AD, Ullian L, Richards WD, Cooper P. Informal advice and information seeking between physicians. Med Educ 1981;56: 174-180.

Wood D. Controlling the medical information baloon. Can Med Assoc J 1982;127-349.

Woodsworth A, Neufeld V. A survey of physician self-education patterns in Toronto, Canada. Part 1: Use of libraries. Can Lib J 1972;29:38-44.

Woolsey FM. Two years of experience with two-way radio conferences for postgraduate medical education. J Med Educ 1958; 33:474-482.

LE RÉSEAU I.S.S.T.: SANTÉ ET SÉCURITÉ DU TRAVAIL AU QUÉBEC

S. L'HEUREUX
CHEF DE SERVICE
J.-P. ARSENAULT
BIBLIOTHÉCAIRE
CENTRE DE DOCUMENTATION
COMMISSTION DE LA SANTÉ ET DE LA SÉCURITÉ DU TRAVAIL
GOUVERNEMENT DU QUÉBEC
MONTRÉAL, QUÉBEC.

RÉSUMÉ

Le réseau I.S.S.T. regroupe divers centres de documentation intéressés à diffuser l'information en santé et sécurité du travail. Les concepteurs du réseau ont misé sur la création d'une base collective de données utilisant le progiciel ISIS. Cette base est interrogée par des utilisateurs qui ne sont pas spécialisés en bibliothéconomie. Cet article fait état des choix du gestionnaire et des résultats constatés.

ABSTRACT

The I.S.S.T. Network includes documentation Centers interested in disseminating information on occupational safety and health. To succeed in this task, the network designers created a cooperative online system using the ISIS software consulted by non-librarians. This paper outlines choices made by the administrator and the results encountered.

LE RÉSEAU I.S.S.T.

INTRODUCTION

" Qu'est-ce que le réseau I.S.S.T. et en quoi diffère-t-il des autres?" . Voilà à peu de chose près ce à quoi nous allons tenter de répondre.

Notre communication comporte cinq parties: d'abord qu'est-ce que la base de données Information en santé et sécurité du travail et quel est son contenu. Nous tenterons ensuite de retracer son historique, puis nous traiterons du réseau I.S.S.T. et de ses particularités. Jean-Pierre Arsenault vous entretiendra ensuite de l'aspect informatique à la Commission de la santé et de la sécurité du travail et vous fera part finalement de nos projets à court et à moyen termes.

LA BASE DE DONNÉES I.S.S.T.

Il nous faut tout de suite préciser que la base I.S.S.T. est un répertoire bibliographique collectif de documentation québécoise en santé et sécurité du travail.

Quoiqu'il s'agisse d'un répertoire bibliographique, I.S.S.T. demeure une base de données et non exclusivement un " répertoire catalographique" .

En effet l'objectif du Centre de documentation de la C.S.S.T. a été double: répertorier la documentation en santé et sécurité du travail et aussi créer un dépôt d'informations bibliographiques facilement utilisables. Le fichier conventionnel de bibliothèque a donc été rejeté au profit de la conception d'un outil de travail plus efficace. Pour ce faire, on a mis l'accent sur le repérage de l'information en utilisant des points d'accès simples et un vocabulaire normalisé.

Bien que I.S.S.T. demeure un répertoire collectif de centres de documentation, nous avons quand même l'impression de créer un " produit" qui dépasse les intérêts immédiats de nos membres. Pour ce faire des règles d'acquisition et de traitement strictes ont été définies.

Quelques considérations géographiques

Le réseau I.S.S.T. compte 19 localisations réparties à travers le Québec. 13 sont à la C.S.S.T. (à Québec, Montréal et dans les bureaux régionaux), 4 dans les Associations sectorielles paritaires et 2 à l'Institut de recherche en santé et sécurité du travail. Il s'agit donc des 19 collections de documents répertoriés dans I.S.S.T. Ceci constitue une facette de son originalité: I.S.S.T. est un répertoire de documents disponibles.

Son contenu

Présentement y sont répertoriés nos 500 abonnements de périodiques et nos 27,000 documents (65% provenant de notre Centre). La couverture est presqu'exclusivement d'ordre scientifique et technique et naturellement son champ d'intérêt couvre la santé et la sécurité du travail.

Nos règles d'acquisition précisent que nous devons inclure en priorité la documentation québécoise et canadienne. Vient, ensuite, avec certaines restrictions, celle des États-Unis, des pays scandinaves, de la France, du Royaume-Uni et de tout autre pays. Puisque cette documentation est d'abord destinée à notre clientèle nous n'incluons que les documents écrits en français (30%) et en anglais (70%).

C'est ce qui fait que notre base de données est différente de la base C.I.S. du Centre international de sécurité et d'hygiène du travail du Bureau international du travail laquelle est polyglotte (moins de 50% du contenu étant en anglais ou en français). On trouve aussi dans I.S.S.T. de la documentation audiovisuelle, particulièrement en français.

Les documents répertoriés sont des monographies, des rapports, des répertoires statistiques, des tirés-à-part etc. Cependant ne pouvant dépouiller systématiquement nos 500 périodiques nous choisissons les principaux pour inclure leurs articles dans notre base. Nos partenaires participent à cet effort en indexant quelques périodiques spécialisés dans leurs domaines. Notre objectif est de partager nos ressources et de parvenir à un dépouillement le plus complet possible. Grâce à cette coopération 37% de nos 27,000 documents sont des articles de périodique.

Diffusion

I.S.S.T. n'est pas encore diffusée par un serveur commercial. Cependant un projet en ce sens est à l'étude.

On ne saurait toutefois passer sous silence deux autres bases de données que produit le Centre de documentation: il s'agit de la base THES et du Système d'information juridique. Nous y reviendrons lorsque nous parlerons du système informatique de la C.S.S.T.

Pour situer cette base dans son contexte, permettez-moi un rappel historique du projet I.S.S.T.

L'HISTORIQUE DE LA BASE DE DONNÉES I.S.S.T.

Le mandat qui nous fut confié à l'automne 1979 précisait la nécessité de mettre sur pied des services documentaires capables de répondre aux besoins en information scienfifique et technique de tous les intervenants auxquels la Loi sur la santé et la sécurité du travail attribuait un rôle.

La Loi sur la santé et la sécurité du travail

La loi qui fut adoptée en décembre 1979 entrait en vigueur en mars 1980. Elle transformait l'ancienne Commission des accidents du travail en lui confiant deux nouveaux mandats, soit la prévention et l'inspection. Ces deux nouvelles dimensions créaient des besoins en terme de documentation. La fonction inspection était auparavant répartie dans quatres ministères différents tandis que la fonction prévention n'en était qu'à ses premiers balbutiements. Tout était donc à faire en fait de développement de la collection et de moyens de diffusion, de manière à atteindre toutes les personnes appelées à travailler dans ce domaine.

LE RÉSEAU I.S.S.T.

Les intervenants

La Loi sur la santé et la sécurité du travail est une loi ambitieuse, qui prévoit la prise en charge par les employeurs et les travailleurs eux-mêmes, des conditions de santé et de sécurité au travail. De nombreux intervenants sont aussi impliqués dans le processus en tant que personnes ressources. D'abord le personnel de la C.S.S.T., chargé de préparer la réglementation et les programmes et de voir à leur application; les équipes de santé au travail faisant partie des Départements de santé communautaire des centres hospitaliers, chargées de la surveillance de la santé dans les établissements; les associations paritaires de prévention dans chacun des secteurs d'activité économique, responsables de la formation et de l'information de leurs membres; les travailleurs et les employeurs eux-mêmes et enfin, le public en général.

Tous ces intervenants ont besoin à des degrés divers d'information scientifique et technique concernant les risques pour la santé et la sécurité dans le milieu de travail, les conséquences de ces risques, les techniques de recherche dans le domaine et les moyens de prévention, afin d'atteindre l'objectif de la loi qui est l'élimination du danger à la source.

Mais tous ces intervenants sont dispersés aux quatre coins du Québec dans 11 bureaux régionaux de la Commission, 32 CH-DSC et 4 associations sectorielles, pour l'instant. Une vingtaine d'associations sectorielles devraient être formées d'ici quelques années.

La situation

Devant ces besoins considérables nous avions deux alternatives: créer un catalogue de bibliothèque conventionnel bien qu'informatisé, en utilisant des vedettes matières ou créer une base de données bibliographiques axée sur les nouvelles tendances des sciences de l'information, utilisant des points d'accès multiples et un vocabulaire normalisé.

Notre clientèle potentielle étant constituée à 90% de non-bibliothécaires, le repérage par de multiples points d'accès et l'utilisation de descripteurs familiers aux spécialistes du domaine nous sont apparus préférables. L'existence du Thésaurus CIS[1] constituait un élément supplémentaire favorisant cette orientation.

Le système informatique

Depuis la régionalisation de la Commission en 1977, l'utilisation d'un terminal à écran cathodique pour consulter les dossiers des accidentés était devenu monnaie courante. On comptait déjà quelques 600 terminaux. Il était donc possible d'envisager la création d'une base de données qui deviendrait accessible aux intervenants de toutes les régions au même titre qu'un autre fichier de la Commission, moyennant une certaine formation.

1- Centre international de sécurité et d'hygiène du travail
Thésaurus CIS, Bureau international du travail, Genève, 1976.

De plus, à cette époque, l'ordinateur n'était pas encore utilisé à son maximum et on nous offrit d'informatiser le nouveau Centre de documentation à condition de ne pas avoir à développer de logiciel.

Le choix du logiciel

Le choix du logiciel ISIS se fit sans trop de problèmes. Ayant vu fonctionner la base de données du Centre de recherches en développement international à Ottawa sur MINISIS et considérant qu'elle répondait à nos besoins, il nous fut facile de remonter au logiciel ISIS qui rencontrait les exigences de notre environnement informatique.

Grâce à ONTERIS, groupe de recherche en éducation du Ministère de l'éducation de l'Ontario, qui utilisait déjà ISIS et à une entente tripartite avec l'Unesco, propriétaire et distributeur du logiciel, nous avons pu recevoir à Toronto la formation nécessaire, au cours de l'été 1980. En janvier 1981, la base de données I.S.S.T. voyait le jour.

LE RÉSEAU I.S.S.T.

Les fournisseurs de données et les utilisateurs

Le Centre de documentation de la C.S.S.T. est à la fois producteur et diffuseur de la base de données I.S.S.T. Il en découle des obligations et quelques fois, des problèmes variés. Il doit coordonner à la fois les activités des fournisseurs de données et celles des simples utilisateurs de la base qui ne sont pas nécessairement fournisseurs. Les exigences des uns et des autres ne coïncident pas toujours et le centre doit concilier les deux sans perdre de vue que son mandat principal consiste à répondre aux besoins d'information documentaire de tout le réseau des intervenants.

Les participants à la base I.S.S.T.

Les participants à la base I.S.S.T. sont au nombre de cinq: l'I.R.S.S.T. et quatre associations sectorielles qui disposent de centres de documentation dotés de personnel ayant une formation en bibliothéconomie et qui développent des collections spécialisées dans leur domaine d'activité. Leur apport est précieux pour la qualité de la base de données puisqu'il assure la couverture de tous les secteurs de façon exhaustive. Bien sûr, cette qualité ne sera atteinte que lorsque toutes les associations sectorielles paritaires seront en fonction. Nous avons donc potentiellement plus de vingt-cinq fournisseurs de données.

Au départ le réseau des fournisseurs a été conçu comme un réseau centralisé afin d'assurer le contrôle de la qualité. Ainsi, la saisie se fait exclusivement au Centre de documentation de la Commission quoiqu'elle aurait pu être effectuée par chacun des partenaires.

Nous avons établi des règles strictes quant à l'inclusion ou non de certains documents. Des échantillonages de contrôle de qualité dans la base nous ont prouvé que nos partenaires partagent nos objectifs.

Nous avons fonctionné pendant plusieurs mois avec nos partenaires sur la base d'accords verbaux. Mais il nous est vite apparu que nous devrions systématiser notre mode de fonctionnement, ce qui a été fait au cours de la dernière année.

Des bordereaux ont été mis au point pour faciliter et uniformiser la transmission des données de saisie.

Les procédures de fonctionnement entre les membres du réseau sont désormais consignées dans un manuel. Les règles de catalogage propres à la base et les politiques d'indexation ont été définies et diffusées.

Les partenaires se sont entendus sur des principes généraux et sur des politiques de développement des collections laissant place aux besoins spécifiques de chacun.

Enfin, pour officialiser ce mode de fonctionnement, nous avons mis au point un protocole d'entente établissant les droits et obligations de chacun des fournisseurs.

Toutes ces étapes ont signifié des consultations auprès de nos partenaires et des séances d'information et de formation.

La création de la base de données du thésaurus découle aussi directement des besoins des indexeurs du réseau. Ceux-ci peuvent transmettre des termes candidats et participent au choix des nouveaux descripteurs. Dans certains cas les indexeurs ont contribué à l'amélioration du thésaurus en développant des facettes qui ne répondaient pas à la nature spécialisée de leur documentation. Ce fut le cas du chapitre "Mesures et recherches" développé par les bibliothécaires de l'I.R.S.S.T.

Les obligations du Centre de documentation

Cette expérience illustre bien les aspects positifs du fonctionnement en réseau. Mais ce mode de fonctionnement crée aussi plusieurs obligations et un certain nombre de problèmes pour le Centre de documentation de Montréal qui en assure la coordination.

Le Centre doit faire le marketing de la base de données. Il doit convaincre les participants éventuels qu'il y a avantage à fonctionner en réseau et doit mettre l'accent sur l'aspect coopératif du système. Ceci est d'autant plus facile que le fichier collectif informatisé élimine la nécessité pour les participants de maintenir des fichiers individuels.

Pour garantir l'homogénéité de la base de données et obtenir un fonctionnement harmonieux, le Centre doit assurer la formation des catalogueurs et des indexeurs. Il doit créer des moyens de transmission de l'information pour faire connaître d'une part les nouveautés concernant le système et d'autre part, pour consulter les participants quant à leurs besoins. Il doit prévoir des réunions de travail fréquentes particulièrement avec les indexeurs.

Le Centre doit également fournir le soutien technique nécessaire à la mise sur pied des centres de documentation du réseau, ce qui génère un grand nombre de communications de part et d'autre.

Enfin, le Centre doit faire preuve de souplesse pour tenir compte des besoins de chaque participant tout en assurant la coordination des activités et en imposant des règles strictes de fonctionnement afin d'assurer un contrôle adéquat.

Les utilisateurs

En plus des participants à la base I.S.S.T. qui sont des fournisseurs de données et des utilisateurs, nous avons aussi un large public susceptible d'interroger la base de données.

Les utilisateurs actuels sont les 11 directions régionales de la Commission, l'I.R.S.S.T., les équipes de santé au travail des 32 départements de santé communautaire, les 6 associations sectorielles paritaires existantes et un certain nombre d'employeurs qui sont reliés à l'ordinateur de la Commission pour l'échange de données d'indemnisation.

Le Centre de documentation assume auprès des usagers d'I.S.S.T., les responsabilités habituelles d'un diffuseur de base de données, soit la formation, l'information et le marketing.

LA FORMATION À L'INTERROGATION

Depuis deux ans nous avons dispensé environ quarante sessions de formation à plus de deux cent trente participants provenant des régions et organismes énumérés précédemment. Il s'agissait de personnes n'ayant aucune formation préalable à l'interrogation des bases de données bibliographiques. Elles étaient désignées afin d'agir à titre de personne-ressource dans leur milieu de travail. Nous comptions sur elles pour maîtriser l'interrogation dans I.S.S.T., transmettre à d'autres la manière d'interroger ou exécuter pour d'autres leurs recherches.

Notre hypothèse de départ était la suivante: nous traitions avec des spécialistes, connaissant bien le vocabulaire en santé-sécurité et intéressés à devenir des utilisateurs compétents.

Malheureusement, les services et organismes responsables de la désignation des futures personnes ressources ne sont pas toujours conscients des exigences de la tâche. La formation à l'interrogation de nouveaux systèmes est une activité courante à la Commission. Mais les sessions relatives à I.S.S.T. présentent deux particularités qui peuvent dérouter les participants. D'abord la difficulté d'expliquer le thésaurus, le choix du vocabulaire contrôlé, la logique booléenne et les nombreux points d'accès. La deuxième consiste à couvrir en une seule journée (temps normalement alloué pour ce type de formation) une matière aussi abondante.

Les données statistiques disponibles ne sont pas suffisantes pour nous fournir une image exacte des résultats de cette formation. Cependant des statistiques globales d'accès à I.S.S.T. nous permettent de constater une sous-utilisation par nos utilisateurs potentiels. Ces taux de consultation s'expliquent de trois façons:

1- Nous avons connu un roulement de personnel important dans tous le réseau des intervenants en santé-sécurité au cours des premières années d'implantation de la loi. Par conséquent la formation devait toujours être reprise;

2- pour des raisons que nous n'avons pas encore évaluées, les personnes formées à l'interrogation en région n'ont pas toujours retransmis la formation;

3- plusieurs intervenants ont été engagés surtout dans des tâches de planification et de mise en place de nouvelles structures depuis le début. Maintenant qu'ils ont abordé la phase d'application des programmes, nul doute que le besoin de documentation scientifique et technique se fera sentir davantage.

Nous avons donc repris dernièrement une nouvelle ronde de formation à l'intention des Départements de santé communautaire. À la Commission, le cours de formation est offert depuis janvier 1984 à l'ensemble du personnel, par le Service de développement des Ressources humaines. Les sessions sont animées par un bibliothécaire du Centre de documentation. On a dû refuser des participants à chacune des trois sessions annoncées. Grâce à ces cours, nous espérons rejoindre des personnes qui ont besoin de documentation dans leur travail quotidien et qui deviendront des interrogateurs assidus des bases de données du Centre de documentation.

Nous espérons qu'en 1984, nous pourrons bénéficier de statistiques plus complètes sur l'accès à la base de données I.S.S.T. et sur l'impact de notre formation.

En conclusion nous pouvons tracer un double bilan de la situation. Nous pensons avoir rencontré nos objectifs au niveau de la qualité du contenu de la base de données I.S.S.T. et de notre fonctionnement en réseau. L'intérêt manifesté par nos partenaires et le taux de croissance de la base au cours de la dernière année le démontrent largement.

Cependant la demi-réussite quant à l'utilisation du système par les usagers potentiels illustre bien les erreurs à éviter: mal identifier la clientèle cible, ne pas assurer un marketing et une information suivis et donner la formation à un groupe d'usagers qui n'en a pas encore ressenti le besoin.

Cependant, à cette étape, toutes nos énergies étaient consacrées à notre clientèle privilégiée des fournisseurs de données.

À quelques reprises, nous avons songé à nous en tenir à la production des bases de données et à confier à un serveur le soin de la diffusion. Cela permettrait d'élargir l'éventail de nos utilisateurs et de répondre à un besoin exprimé par différents milieux, particulièrement le milieu universitaire et collégial. À court terme cependant et malgré les difficultés, nous estimons que la formation des utilisateurs des bases de données du Centre est fondamentale et doit se poursuivre.

L'ASPECT INFORMATIQUE DE LA C.S.S.T.

La quincaillerie

Avant de parler du progiciel ISIS que nous utilisons, peut-être pouvons-nous dire quelques mots du système informatique de la C.S.S.T.

localisé à Québec.

Ce système se compose tout d'abord de deux ordinateurs I.B.M. 3081 de modèles G et K. Ces deux appareils se partagent le développement et la production. Ils ont une capacité de mémoire centrale de vingt mega-octets, tandis que la mémoire auxiliaire se compose de vingt unités de disque 3380, de cinq unités 3350 et d'une douzaine de lecteurs de ruban.

Pour communiquer avec ses 1,300 périphériques le réseau utilise deux contrôleurs centraux qui eux utilisent un réseau de contrôleurs de type 3274 en région, via des lignes dédiées.

Les logiciels

Au niveau logiciel les communications sont régies par VTAM et NETMON, deux logiciels de communication et de contrôle de réseau.

ISIS est exécuté sous C.I.C.S. (Customer Information Control System) qui lui-même est exécuté sous MVS-XA (système d'exploitation). Le fonctionnement adéquat d'ISIS dépend de la compatibilité des composantes du système (quincaillerie et logiciels). Il est arrivé assez fréquemment qu'un simple changement d'unité de disque ou de version de système d'exploitation bouleverse l'exécution des travaux quotidiens du progiciel ISIS.

Une bonne communication entre les utilisateurs locaux et l'équipe de développement et d'entretien informatique s'avère essentielle. Cependant le progiciel ISIS étant non-conventionnel pour les informaticiens habitués aux produits I.B.M., nous aurions pu rencontrer plus de problèmes que nous n'en n'avons éprouvés en réalité. Il nous faut remercier monsieur G. Del Bigio, Chef Adjoint, Division de la bibliothèque de l'UNESCO des archives et des services documentaires et son équipe, qui nous ont donné un " service après vente" fort apprécié.

Le logiciel ISIS

La version 3.2 d'ISIS ou Integrated Scientific Information System a été installé à la C.S.S.T. fin 1980 grâce à la collaboration du groupe ONTERIS lui-même utilisateur du progiciel.

Comme son nom l'indique, ISIS est un système intégré de base de données qui permet au non-initié de l'informatique de gérer facilement son système. Sur la fiche technique ci-jointe vous pouvez constater certaines limitations imposées par le progiciel. Lorsque l'information était disponible nous avons ajouté la modification apportée par la version 4.4. De ces modifications soulignons surtout le nombre d'enregistrements par base de données et la taille de ceux-ci qui ont été considérablement augmentés. Il existe présentement deux modules qui permettent les opérations de saisie et de recherche. Cependant la version 4.4 que nous sommes en train d'installer, ne possède qu'une seule transaction permettant, par exemple, aux opérateurs autorisés, la modification de données en cours de recherche.

La structure de ce progiciel permet au non-initié de gérer et définir facilement sa base de données. Cette définition est enregistrée dans un fichier appelé F.D.T. (Field Definition Table) où le documentaliste inscrit les paramètres de la base de données. Vous trouverez ci-joint un extrait du F.D.T. de la base de données I.S.S.T. Un autre fichier le F.S.T. (Field Select Table) contient les méthodes d'extraction et les tris.

Outre ces deux fichiers soulignons qu'il en existe plusieurs autres dont le fichier maître, le fichier inversé (comprenant le fichier d'index) et le fichier des transactions.

Règle générale, la recherche est exécutée en direct et les opérations de mise à jour, en différé. Nous utilisons le module de saisie en direct mais ne faisons que deux mises à jour par semaine.

Il existe aussi un module de recherche en différé (batch retrieval) qui nous sert à produire soit des listes de gestion, soit des bibliographies.

Les autres systèmes du Centre de documentation

Nous ne saurions passer sous silence deux autres systèmes développés par notre Centre bien qu'à eux seuls ils pourraient faire l'objet d'une communication. La base THES et le système BIJ (Banque d'information juridique) sont diffusés à tous les utilisateurs du réseau I.S.S.T.

La base de données THES

La base de données THES est en fait la version informatique du Thésaurus C.S.S.T. Nous avons choisi d'utiliser le logiciel ISIS pour créer le thésaurus normalisant le vocabulaire servant à l'indexation des documents repertoriés dans la base I.S.S.T. La version 3.2 ne possédantpas de module de gestion de thésaurus et devant l'urgence d'avoir un outil facile de mise à jour diffusable sur le réseau, nous avons décidé de créer ce thésaurus sous forme de base de données. D'ailleurs un extrait du F.D.T. se trouve en annexe.

La base THES sert aussi à la validation automatisée de l'indexation de I.S.S.T. Ce thésaurus qui a pour ancêtre le Thésaurus CIS possède plus de 13,000 descripteurs et expressions rejetées. Notons aussi que le Thésaurus C.S.S.T. est imprimé sur imprimante au laser via le module de photocomposition d'ISIS.

Le système BIJ

Le système des Banques d'information juridique contient le texte intégral des lois et règlements administrés par la C.S.S.T. de même que des références à la jurisprudence émise par diverses autorités en santé et sécurité du travail.

Les BIJ utilisent le progiciel STAIRS pour le repérage et ATMS pour la saisie. Ce système n'étant pas intégré, cela crée divers problèmes de gestion et d'utilisation quenous ne traiterons pas ici. Cependant, les intéressés trouveront en annexe un diagramme de fonctionnement.

LES DÉVELOPPEMENTS FUTUR

À court et à moyen termesles développements futurs comprennent l'implantation de la version 4.4 du progiciel ISIS, l'automatisation des acquisitions de notre Centre et l'utilisation de la micro-informatique.

La version 4.4 d'ISIS

Fin 1984 nous comptons avoir terminé l'implantation de la version 4.4 d'ISIS. Le passage de 3.2 à 4.4 permettra de résoudre certains problèmes que nous avons rencontrés tels la longueur des points d'accès qui passe de 30 à 256 caractères, la conversion des fichiers ISAM à VSAM, le nombre d'enregistrements maximal par base de données passant de 100,000 à 16,000,000, et la limite du nombre d'enregistrements repérables simultanément qui est éliminée.

À ces considérations s'ajoutent les capacités nouvelles du programme qui valent la peine à elles seules de changer de version: deux types de modification globale, un éditeur " full screen editor" , la gestion améliorée du dictionnaire, une transaction unique, de nouveaux opérateurs logiques de recherche, la sécurité accrue des données, la possibilité de créer des enregistrements dits " virtuels" , l'amélioration considérable du " Print formatting language" , la possibilité de créer des routines internes de type conditionnel (if...then) pour la validation ou la création d'enregistrements et finalement la création d'un module d'acquisition dont on peut varier les paramètres.

Ces nombreuses possibilités nous permettront sans doute de faciliter et d'améliorer notre gestion de la base de données et d'offrir des services encore plus adaptés aux besoins de nos partenaires.

L'automatisation des acquisitions

La possibilité d'automatiser nos procédures d'acquisition via ISIS a été un autre facteur influançant notre décision de changer de version. Notre Centre ayant un volume d'acquisition de monographies et de périodiques dépassant 4,000 par année et afin d'améliorer notre performance et d'augmenter l'information de gestion sur ce système, l'automatisation nous est apparue comme la solution.

ISIS nous permettra d'intégrer encore plus nos opérations et rééquilibrera certaines tâches de la " chaîne documentaire" du Centre. L'information trouvée et saisie aux acquisitions servira désormais de pré-catalogage et de pré-saisie. Ces opérations seront implantées, dans un premier temps, à notre Centre de documentation et à l'I.R.S.S.T. Cependant son utilisation par nos partenaires pourrait être envisagée sans mettre en cause la conception du réseau centralisé ayant à sa base le contrôle de la qualité.

À moyen terme, le Centre de documentation espère faire l'acquisition de ressources en micro-informatique pour le traitement local des prêts entre bibliothèques et pour l'exploration des possibilités de " down loading" d'information scientifique et technique en santé et sécurité du travail provenant de banques consentantes. Cette information épurée et remise en forme pourrait être incluse dans notre base de données ou faire l'objet de fichiers locaux pour consultation restreinte.

Rappelons en terminant, que nous accueillerons parmi nos partenaires, fournisseurs de données, quatre ou cinq nouvelles associations sectorielles au cours des prochains mois.

Enfin, tel que mentionné plus tôt, nous accentuerons nos efforts de formation auprès des utilisateurs potentiels des bases de données du Centre de documentation.

MIPS to the People

Richard Larratt, M.A., P.Eng.
R.R. #1 Demorestville, Ont.
Canada, **KOK 1WO**

Abstract

Many writers suggest that computers enable the mega-state and "1984"-like isolate and over-power the individual. Such writers cite examples of citizen harassment by well-funded well-equipped bureaucracy. These fears were played to in a recent Revenue Taxation Film: "Excuse Me, There is a Computer Watching You". This writer presents the counter-point and counter-reaction.

MIPS aux gens!

Nombre d'auteurs suggèrent que les ordinateurs favorisent l'état tout puissant et, ainsi que le roman «1984» le décrit, lui permettent d'isoler et d'écraser l'individu. On parle ainsi du harcèlement du citoyen par la bureaucratie bien pourvue en fonds et matériel. Ces craintes ont été renforcées par un récent film de Revenu Canada: «Excusez-moi, un ordinateur vous demande». Cet auteur présente le contrepoint et les contre-réactions dans ce domaine.

Introduction

Perhaps because this is "1984" we are more aware this year, than any year after '48 of just how far computers have tipped power in the direction of the large organization ... be that organization one of state or corporation.

Other writers have noted that Orwell did not really mention computers "as such" in his book. Yet it does not take much imagination to see how willing and useful computers, in particular, the database sort we deal with everyday as research librarians, might be in providing an "evil" state with the necessary information stuff for "Orwellian" control.

Since the time and input of Einstein and Turing we have seen the world transformed by nuclear bombs and computers. I will place the bombs in the "space" as opposed to the "information" domain but even after twenty years in the computer and information business I am not that comfortable with this distinction between "space" and "information" ... yet who would not admit that the violence of Hiroshima was "spacial" whereas the assault on my privacy by computers is an assault of my information domain. For this writer's eye these words "limp"; I trust the reader "knowing what I mean".

In the Beginning

At the onset of the nuclear age, the U.S.A. by trial and persecution of it's scientists, see Oppenheimer, did what it could to delay the spread of nuclear weapons. When I was working on the war gaming computers inside the Defense Research Board more than twenty years ago, the Nuclear Club was still less than ten nations. How many is it today? Has the possession of Nuclear Weapons enabled the possessor to either stop or start a revolution or even to have it's own way. How about ten thousand AK-47's and the teenagers to carry them?

At the onset of the computer age, the few computers not in military use were either at universities or entombed in the tax department or a life insurance company . Throughout the sixties it remained so. Sure, computers became cheaper and found more and more application, but it was not until the advent of the personal computer that the possibility for the computer being an instrument of freedom not oppression became apparent.

Big Computers, Big Organizations

In part because this is "1984" and in part because large organizations have exceeded the boundaries of human decency in their mis-use of computers, average folks are increasingly aware of how computers are being used (mis-used) by large organizations in their pursuit of the individual, whether to sell, control or tax.

"Pardon me, there's a computer asking for you", is not a joke, it is the title of a (withdrawn) film Revenue Canada commissioned to frighten people. To the extent that fact and fear merge it might be taken as a sign of how things are proceeding in the bureacratic bowels of Ottawa.

Conversely

What about the converse? If the big guys can have their databases, their telecommunication grids, their LANS (local area networks) what is the chance/hope for the isolated individual. Ans. If over 40, if computer illiterate ... only political or luddite. If under 40, if computer literate ... this writer (43 but computer literate) says "out-compute/out-network them" much as the american revolutionaries "outgunned" the British in their war of independence (the old kentucky rifle versus the british musket story).

The article concludes by brief capitulation of the three options or reactions: political, luddite and "out-gunning".

Political

Freedom of Information. It is simply a national scandal, that the federal government has so well blocked information access in Canada (define "cabinet document") so bias affairs in favour of itself and it's fellow-traveller big business ... that on the occasion of a researcher in the P.M.O.'s looking for dirt on Mulroney ... researcher went to Washington: for Canadian information.

Since Canada makes no claim to being an "imperial" power, since there are no "national" security issues of more import than the fall of the Liberal Party, since that danger to the Nation is the greater of the evils, the other being that some laundry gets washed in the public eye let us proceed towards disclosure. Let's support those politicians that move in our cautious Canadian way to opening up the files. Having seen some of these SECRET things I'd think the greatest danger they pose is that some of population may die laughing.

Luddite

"Office Automation", "Factory of the Future" ... who other than a "Luddite" can oppose the changes these phrases encompass? That's right ... only a "Luddite" ... and they'll be no more successful than were the originals. If the Luddites win ... its GOSUB THIRD WORLD a place most Canadians go only with an Air Canada RETURN.

On the other hand, there is money and opportunity for a few with the will and wit to go against the stream. I have a hand-made leather case for my RADIO SHACK Model 100 (for the Luddites reading the article, that's a lap computer and I paid $200 for the case and $700 for the computer).

Out-compute: Out-network

I've run some big many-MIPS computers in my day and I'll opine that in the information wars now in their openings they're a lot like the nuclear bombs in the big-power arsenals i.e. at the "street" level "who cares" ... in Central America neither Russia or America can "use" those N. weapons ... everyday killing is done with American M-16's, Russian AK-47's, knives, clubs, et cetera.

It is important to note that most of the 3 million plus personal computers being sold this year in N.A. are going into big not small business.

Writer, Guru James Martin says it all if you care to pay the $1000 or more for attendance at one of his seminars ... otherwise read his Information Manifesto and realize that what he is talking (big computers, big database systems) ... don't work: yet. Of course they have to and that's why corporate America keeps going to his Seminars. Come to mind. There's a better answer aborning in the new computer literacy.

AUTHOR INDEX/INDEX DES AUTEURS

Arsenault, Jean Pierre 178

Barker, J.L. 29

Benedetti, J.L. 56

Blackie, Raymond G. 36

Bodorik, P. 89

Campbell, Mary E. 114

Cohen, Charles P. 150

Crawford, R.G. 70

Doll, L.F. "Buddy" 114

Duchesne, Roddy 142

Gallichand, S. 56

Grusec, T. 9

Johnston, A.W. 23

Larratt, Richard 190

Lefebvre, L. 56

L'Heureux, Suzanne 178

MacLeod, Ian 77

Meadow, Charles T. 1

Nash, John C. 133

Nash, Mary M. 133

Owen, Victor M. 156

Park, N. 9

Sanfaçon, G. 56

Shepherd, Michael A. 45

Warren, Joanne L. 163

Weber, J.P. 56